Lecture Notes in Computer Science 3046

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

Takeo Kanade
Carnegie Mellon University, Pittsburgh, PA, USA
Josef Kittler
University of Surrey, Guildford, UK
Jon M. Kleinberg
Cornell University, Ithaca, NY, USA
Friedemann Mattern
ETH Zurich, Switzerland
John C. Mitchell
Stanford University, CA, USA
Oscar Nierstrasz
University of Bern, Switzerland
C. Pandu Rangan
Indian Institute of Technology, Madras, India
Bernhard Steffen
University of Dortmund, Germany
Madhu Sudan
Massachusetts Institute of Technology, MA, USA
Demetri Terzopoulos
New York University, NY, USA
Doug Tygar
University of California, Berkeley, CA, USA
Moshe Y. Vardi
Rice University, Houston, TX, USA
Gerhard Weikum
Max-Planck Institute of Computer Science, Saarbruecken, Germany

Springer
Berlin
Heidelberg
New York
Hong Kong
London
Milan
Paris
Tokyo

Antonio Laganà Marina L. Gavrilova
Vipin Kumar Youngsong Mun
C.J. Kenneth Tan Osvaldo Gervasi (Eds.)

Computational Science and Its Applications - ICCSA 2004

International Conference
Assisi, Italy, May 14-17, 2004
Proceedings, Part IV

Springer

Volume Editors

Antonio Laganà
University of Perugia, Department of Chemistry
Via Elce di Sotto, 8, 06123 Perugia, Italy
E-mail: lag@unipg.it

Marina L. Gavrilova
University of Calgary, Department of Computer Science
2500 University Dr. N.W., Calgary, AB, T2N 1N4, Canada
E-mail: marina@cpsc.ucalgary.ca

Vipin Kumar
University of Minnesota, Department of Computer Science and Engineering
4-192 EE/CSci Building, 200 Union Street SE, Minneapolis, MN 55455, USA
E-mail: kumar@cs.umn.edu

Youngsong Mun
SoongSil University, School of Computing, Computer Communication Laboratory
1-1 Sang-do 5 Dong, Dong-jak Ku, Seoul 156-743, Korea
E-mail: mun@computing.soongsil.ac.kr

C.J. Kenneth Tan
Queen's University Belfast, Heuchera Technologies Ltd.
Lanyon North, University Road, Belfast, Northern Ireland, BT7 1NN, UK
E-mail: cjtan@optimanumerics.com

Osvaldo Gervasi
University of Perugia, Department of Mathematics and Computer Science
Via Vanvitelli, 1, 06123 Perugia, Italy
E-mail: ogervasi@computer.org

Library of Congress Control Number: 2004105531

CR Subject Classification (1998): D, F, G, H, I, J, D.2-3

ISSN 0302-9743
ISBN 3-540-22060-7 Springer-Verlag Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable to prosecution under the German Copyright Law.

Springer-Verlag is a part of Springer Science+Business Media

springeronline.com

© Springer-Verlag Berlin Heidelberg 2004
Printed in Germany

Typesetting: Camera-ready by author, data conversion by PTP-Berlin, Protago-TeX-Production GmbH
Printed on acid-free paper SPIN: 11010128 06/3142 5 4 3 2 1 0

Preface

The natural mission of Computational Science is to tackle all sorts of human problems and to work out *intelligent* automata aimed at alleviating the burden of working out suitable tools for solving complex problems. For this reason Computational Science, though originating from the need to solve the most challenging problems in science and engineering (computational science is the key player in the fight to gain fundamental advances in astronomy, biology, chemistry, environmental science, physics and several other scientific and engineering disciplines) is increasingly turning its attention to all fields of human activity.

In all activities, in fact, intensive computation, information handling, knowledge synthesis, the use of ad-hoc devices, etc. increasingly need to be exploited and coordinated regardless of the location of both the users and the (various and heterogeneous) computing platforms. As a result the key to understanding the explosive growth of this discipline lies in two adjectives that more and more appropriately refer to Computational Science and its applications: interoperable and ubiquitous. Numerous examples of ubiquitous and interoperable tools and applications are given in the present four LNCS volumes containing the contributions delivered at the 2004 International Conference on Computational Science and its Applications (ICCSA 2004) held in Assisi, Italy, May 14–17, 2004.

To emphasize this particular connotation of modern Computational Science the conference was preceded by a tutorial on Grid Computing (May 13–14) concertedly organized with the COST D23 Action (METACHEM: Metalaboratories for Complex Computational Applications in Chemistry) of the European Coordination Initiative COST in Chemistry and the Project *Enabling Platforms for High-Performance Computational Grids Oriented to Scalable Virtual Organization* of the Ministry of Science and Education of Italy.

The volumes consist of 460 peer reviewed papers given as oral contributions at the conference. The conference included 8 presentations from keynote speakers, 15 workshops and 3 technical sessions. Thanks are due to most of the workshop organizers and the Program Committee members, who took care of the unexpected exceptional load of reviewing work (either carrying it out by themselves or distributing it to experts in the various fields).

Special thanks are due to Noelia Faginas Lago for handling all the necessary secretarial work. Thanks are also due to the young collaborators of the High Performance Computing and the Computational Dynamics and Kinetics research groups of the Department of Mathematics and Computer Science and of the Department of Chemistry of the University of Perugia. Thanks are, obviously,

due as well to the sponsors for supporting the conference with their financial and organizational help.

May 2004

Antonio Laganà
on behalf of the co-editors:
Marina L. Gavrilova
Vipin Kumar
Youngsong Mun
C.J. Kenneth Tan
Osvaldo Gervasi

Organization

ICCSA 2004 was organized by the University of Perugia, Italy; the University of Minnesota, Minneapolis (MN), USA and the University of Calgary, Calgary (Canada).

Conference Chairs

Osvaldo Gervasi (University of Perugia, Perugia, Italy), Conference Chair
Marina L. Gavrilova (University of Calgary, Calgary, Canada),
Conference Co-chair
Vipin Kumar (University of Minnesota, Minneapolis, USA), Honorary Chair

International Steering Committee

J.A. Rod Blais (University of Calgary, Canada)
Alexander V. Bogdanov (Institute for High Performance Computing and Data Bases, Russia)
Marina L. Gavrilova (University of Calgary, Canada)
Andres Iglesias (University de Cantabria, Spain)
Antonio Laganà (University of Perugia, Italy)
Vipin Kumar (University of Minnesota, USA)
Youngsong Mun (Soongsil University, Korea)
Reneé S. Renner (California State University at Chico, USA)
C.J. Kenneth Tan (Heuchera Technologies, Canada and The Queen's University of Belfast, UK)

Local Organizing Committee

Osvaldo Gervasi (University of Perugia, Italy)
Antonio Laganà (University of Perugia, Italy)
Noelia Faginas Lago (University of Perugia, Italy)
Sergio Tasso (University of Perugia, Italy)
Antonio Riganelli (University of Perugia, Italy)
Stefano Crocchianti (University of Perugia, Italy)
Leonardo Pacifici (University of Perugia, Italy)
Cristian Dittamo (University of Perugia, Italy)
Matteo Lobbiani (University of Perugia, Italy)

Workshop Organizers

Information Systems and Information Technologies (ISIT)

Youngsong Mun (Soongsil University, Korea)

Approaches or Methods of Security Engineering

Haeng Kon Kim (Catholic University of Daegu, Daegu, Korea)
Tai-hoon Kim (Korea Information Security Agency, Korea)

Authentication Technology

Eui-Nam Huh (Seoul Women's University, Korea)
Ki-Young Mun (Seoul Women's University, Korea)
Taemyung Chung (Seoul Women's University, Korea)

Internet Communications Security

José Sierra-Camara (ITC Security Lab., University Carlos III of Madrid, Spain)
Julio Hernandez-Castro (ITC Security Lab., University Carlos III of Madrid, Spain)
Antonio Izquierdo (ITC Security Lab., University Carlos III of Madrid, Spain)

Location Management and Security in Next Generation Mobile Networks

Dong Chun Lee (Howon University, Chonbuk, Korea)
Kuinam J. Kim (Kyonggi University, Seoul, Korea)

Routing and Handoff

Hyunseung Choo (Sungkyunkwan University, Korea)
Frederick T. Sheldon (Sungkyunkwan University, Korea)
Alexey S. Rodionov (Sungkyunkwan University, Korea)

Grid Computing

Peter Kacsuk (MTA SZTAKI, Budapest, Hungary)
Robert Lovas (MTA SZTAKI, Budapest, Hungary)

Resource Management and Scheduling Techniques for Cluster and Grid Computing Systems

Jemal Abawajy (Carleton University, Ottawa, Canada)

Parallel and Distributed Computing

Jiawan Zhang (Tianjin University, Tianjin, China)
Qi Zhai (Tianjin University, Tianjin, China)
Wenxuan Fang (Tianjin University, Tianjin, China)

Molecular Processes Simulations

Antonio Laganà (University of Perugia, Perugia, Italy)

Numerical Models in Biomechanics

Jiri Nedoma (Academy of Sciences of the Czech Republic, Prague, Czech Republic)
Josef Danek (University of West Bohemia, Pilsen, Czech Republic)

Scientific Computing Environments (SCEs) for Imaging in Science

Almerico Murli (University of Naples Federico II and Institute for High Performance Computing and Networking, ICAR, Italian National Research Council, Naples, Italy)
Giuliano Laccetti (University of Naples Federico II, Naples, Italy)

Computer Graphics and Geometric Modeling (TSCG 2004)

Andres Iglesias (University of Cantabria, Santander, Spain)
Deok-Soo Kim (Hanyang University, Seoul, Korea)

Virtual Reality in Scientific Applications and Learning

Osvaldo Gervasi (University of Perugia, Perugia, Italy)

Web-Based Learning

Woochun Jun (Seoul National University of Education, Seoul, Korea)

Matrix Approximations with Applications to Science, Engineering and Computer Science

Nicoletta Del Buono (University of Bari, Bari, Italy)
Tiziano Politi (Politecnico di Bari, Bari, Italy)

Spatial Statistics and Geographic Information Systems: Algorithms and Applications

Stefania Bertazzon (University of Calgary, Calgary, Canada)
Borruso Giuseppe (University of Trieste, Trieste, Italy)

Computational Geometry and Applications (CGA 2004)

Marina L. Gavrilova (University of Calgary, Calgary, Canada)

Program Committee

Jemal Abawajy (Carleton University, Canada)
Kenny Adamson (University of Ulster, UK)
Stefania Bertazzon (University of Calgary, Canada)
Sergei Bespamyatnikh (Duke University, USA)
J.A. Rod Blais (University of Calgary, Canada)
Alexander V. Bogdanov (Institute for High Performance Computing and Data Bases, Russia)
Richard P. Brent(Oxford University, UK)
Martin Buecker (Aachen University, Germany)
Rajkumar Buyya (University of Melbourne, Australia)
Hyunseung Choo (Sungkyunkwan University, Korea)
Toni Cortes (Universidad de Catalunya, Barcelona, Spain)
Danny Crookes (The Queen's University of Belfast, (UK))
Brian J. d'Auriol (University of Texas at El Paso, USA)
Ivan Dimov (Bulgarian Academy of Sciences, Bulgaria)
Matthew F. Dixon (Heuchera Technologies, UK)
Marina L. Gavrilova (University of Calgary, Canada)
Osvaldo Gervasi (University of Perugia, Italy)
James Glimm (SUNY Stony Brook, USA)
Christopher Gold (Hong Kong Polytechnic University, Hong Kong, ROC)
Paul Hovland (Argonne National Laboratory, USA)
Andres Iglesias (University de Cantabria, Spain)
Elisabeth Jessup (University of Colorado, USA)
Chris Johnson (University of Utah, USA)
Peter Kacsuk (Hungarian Academy of Science, Hungary)
Deok-Soo Kim (Hanyang University, Korea)
Vipin Kumar (University of Minnesota, USA)
Antonio Laganà (University of Perugia, Italy)
Michael Mascagni (Florida State University, USA)
Graham Megson (University of Reading, UK)
Youngsong Mun (Soongsil University, Korea)
Jiri Nedoma (Academy of Sciences of the Czech Republic, Czech Republic)
Robert Panoff (Shodor Education Foundation, USA)
Reneé S. Renner (California State University at Chico, USA)
Heather J. Ruskin (Dublin City University, Ireland)
Muhammad Sarfraz (King Fahd University of Petroleum and Minerals, Saudi Arabia)
Edward Seidel (Louisiana State University, (USA) and Albert-Einstein-Institut, Potsdam, Germany)
Vaclav Skala (University of West Bohemia, Czech Republic)
Masha Sosonkina (University of Minnesota, (USA))
David Taniar (Monash University, Australia)
Ruppa K. Thulasiram (University of Manitoba, Canada)
Koichi Wada (University of Tsukuba, Japan)

Stephen Wismath (University of Lethbridge, Canada)
Chee Yap (New York University, USA)
Osman Yaşar (SUNY at Brockport, USA)

Sponsoring Organizations

University of Perugia, Perugia, Italy

University of Calgary, Calgary, Canada

University of Minnesota, Minneapolis, MN, USA

The Queen's University of Belfast, UK

Heuchera Technologies, UK

The project **GRID.IT**: *Enabling Platforms for High-Performance Computational Grids Oriented to Scalable Virtual Organizations*, of the Ministry of Science and Education of Italy

COST – European Cooperation in the Field of Scientific and Technical Research

Table of Contents – Part IV

Track on Numerical Methods and Algorithms

New Techniques in Designing Finite Difference Domain Decomposition Algorithm for the Heat Equation 1
Weidong Shen, Shulin Yang

A Fast Construction Algorithm for the Incidence Matrices of a Class of Symmetric Balanced Incomplete Block Designs 11
Ju-Hyun Lee, Sungkwon Kang, Hoo-Kyun Choi

ILUTP_Mem: A Space-Efficient Incomplete LU Preconditioner 20
Tzu-Yi Chen

Optimal Gait Control for a Biped Locomotion Using Genetic Algorithm 29
Jin Geol Kim, SangHo Choi, Ki heon Park

A Bayes Algorithm for the Multitask Pattern Recognition Problem – Direct and Decomposed Independent Approaches 39
Edward Puchala

Energy Efficient Routing with Power Management to Increase Network Lifetime in Sensor Networks 46
Hyung-Wook Yoon, Bo-Hyeong Lee, Tae-Jin Lee, Min Young Chung

New Parameter for Balancing Two Independent Measures in Routing Path 56
Moonseong Kim, Young-Cheol Bang, Hyunseung Choo

A Study on Efficient Key Distribution and Renewal in Broadcast Encryption 66
Deok-Gyu Lee, Im-Yeong Lee

Track on Parallel and Distributed Computing

Self-Tuning Mechanism for Genetic Algorithms Parameters, an Application to Data-Object Allocation in the Web 77
Joaquín Pérez, Rodolfo A. Pazos, Juan Frausto, Guillermo Rodríguez, Laura Cruz, Graciela Mora, Héctor Fraire

Digit-Serial AB^2 Systolic Array for Division in GF(2^m) 87
Nam-Yeun Kim, Kee-Young Yoo

Design and Experiment of a Communication-Aware Parallel Quicksort with Weighted Partition of Processors 97
Sangman Moh, Chansu Yu, Dongsoo Han

A Linear Systolic Array for Multiplication in $GF(2^m)$ for High Speed Cryptographic Processors 106
Soonhak Kwon, Chang Hoon Kim, Chun Pyo Hong

Price Driven Market Mechanism for Computational Grid Resource Allocation 117
Chunlin Li, Zhengding Lu, Layuan Li

A Novel LMS Method for Real-Time Network Traffic Prediction 127
Yang Xinyu, Zeng Ming, Zhao Rui, Shi Yi

Dynamic Configuration between Proxy Caches within an Intranet 137
Víctor J. Sosa Sosa, Juan G. González Serna, Xochitl Landa Miguez, Francisco Verduzco Medina, Manuel A. Valdés Marrero

A Market-Based Scheduler for JXTA-Based Peer-to-Peer Computing System 147
Tan Tien Ping, Gian Chand Sodhy, Chan Huah Yong, Fazilah Haron, Rajkumar Buyya

Reducing on the Number of Testing Items in the Branches of Decision Trees 158
Hyontai Sug

CORBA-Based, Multi-threaded Distributed Simulation of Hierarchical DEVS Models: Transforming Model Structure into a Non-hierarchical One 167
Ki-Hyung Kim, Won-Seok Kang

The Effects of Network Topology on Epidemic Algorithms 177
Jesús Acosta-Elías, Ulises Pineda, Jose Martin Luna-Rivera, Enrique Stevens-Navarro, Isaac Campos-Canton, Leandro Navarro-Moldes

A Systematic Database Summary Generation Using the Distributed Query Discovery System 185
Tae W. Ryu, Christoph F. Eick

Parallel Montgomery Multiplication and Squaring over GF(2^m) Based on Cellular Automata 196
Kyo Min Ku, Kyeoung Ju Ha, Wi Hyun Yoo, Kee Young Yoo

A Decision Tree Algorithm for Distributed Data Mining: Towards Network Intrusion Detection 206
Sung Baik, Jerzy Bala

Maximizing Parallelism for Nested Loops with Non-uniform Dependences 213
Sam Jin Jeong

Fair Exchange to Achieve Atomicity in Payments of High Amounts Using Electronic Cash 223
Magdalena Payeras-Capella, Josep Lluís Ferrer-Gomila, Llorenç Huguet-Rotger

Gossip Based Causal Order Broadcast Algorithm 233
ChaYoung Kim, JinHo Ahn, ChongSun Hwang

Track on Signal Processing

Intermediate View Synthesis from Stereoscopic Videoconference Images 243
Chaohui Lu, Ping An, Zhaoyang Zhang

Extract Shape from Clipart Image Using Modified Chain Code – Rectangle Representation 251
Chang-Gyu Choi, Yongseok Chang, Jung-Hyun Cho, Sung-Ho Kim

Control Messaging Channel for Distributed Computer Systems 261
Bogusław Cyganek, Jan Borgosz

Scene-Based Video Watermarking for Broadcasting Systems 271
Uk-Chul Choi, Yoon-Hee Choi, Dae-Chul Kim, Tae-Sun Choi

Distortion-Free of General Information with Edge Enhanced Error Diffusion Halftoning 281
Byong-Won Hwang, Tae-Ha Kang, Tae-Seung Lee

Enhanced Video Coding with Error Resilience Based on Macroblock Data Manipulation 291
Tanzeem Muzaffar, Tae-Sun Choi

Filtering of Colored Noise for Signal Enhancement 301
Myung Eui Lee, Pyung Soo Kim

Model-Based Human Motion Tracking and Behavior Recognition Using Hierarchical Finite State Automata 311
Jihun Park, Sunghun Park, J.K. Aggarwal

Effective Digital Watermarking Algorithm by Contour Detection 321
Won-Hyuck Choi, Hye-jin Shim, Jung-Sun Kim

New Packetization Method for Error Resilient Video Communications ... 329
Kook-yeol Yoo

A Video Mosaicking Technique with Self Scene Segmentation for Video Indexing 338
Yoon-Hee Choi, Yeong Kyeong Seong, Joo-Young Kim, Tae-Sun Choi

Real-Time Video Watermarking for MPEG Streams 348
Kyung-Pyo Kang, Yoon-Hee Choi, Tae-Sun Choi

A TCP-Friendly Congestion Control Scheme Using Hybrid Approach for Reducing Transmission Delay of Real-Time Video Stream 359
Jong-Un Yang, Jeong-Hyun Cho, Sang-Hyun Bae, In-Ho Ra

Object Boundary Edge Selection Using Level-of-Detail Canny Edges 369
Jihun Park, Sunghun Park

Inverse Dithering through IMAP Estimation 379
Monia Discepoli, Ivan Gerace

A Study on Neural Networks Using Taylor Series Expansion of Sigmoid Activation Function 389
Fevzullah Temurtas, Ali Gulbag, Nejat Yumusak

A Study on Neural Networks with Tapped Time Delays: Gas Concentration Estimation 398
Fevzullah Temurtas, Cihat Tasaltin, Hasan Temurtas, Nejat Yumusak, Zafer Ziya Ozturk

Speech Emotion Recognition and Intensity Estimation 406
Mingli Song, Chun Chen, Jiajun Bu, Mingyu You

Speech Hiding Based on Auditory Wavelet 414
Liran Shen, Xueyao Li, Huiqiang Wang, Rubo Zhang

Automatic Selecting Coefficient for Semi-blind Watermarking 421
Sung-kwan Je, Jae-Hyun Cho, Eui-young Cha

Track on Telecommunications

Network Probabilistic Connectivity: Optimal Structures 431
Olga K. Rodionova, Alexey S. Rodionov, Hyunseung Choo

Differentiated Web Service System through Kernel-Level Realtime Scheduling and Load Balancing 441
Myung-Sub Lee, Chang-Hyeon Park, Young-Ho Sohn

Adaptive CBT/Anycast Routing Algorithm for Multimedia Traffic Overload 451
Kwnag-Jae Lee, Won-Hyuck Choi, Jung-Sun Kim

Achieving Fair New Call CAC for Heterogeneous Services in Wireless Networks 460
SungKee Noh, YoungHa Hwang, KiIl Kim, SangHa Kim

Track on Visualization and Virtual and Augmented Reality

Application of MCDF Operations in Digital Terrain Model Processing 471
Zhiqiang Ma, Anthony Watson, Wanwu Guo

Visual Mining of Market Basket Association Rules 479
Kesaraporn Techapichetvanich, Amitava Datta

Visualizing Predictive Models in Decision Tree Generation 489
Sung Baik, Jerzy Bala, Sung Ahn

Track on Software Engineering

A Model for Use Case Priorization Using Criticality Analysis 496
José Daniel García, Jesús Carretero, José María Pérez, Félix García

Using a Goal-Refinement Tree to Obtain and Refine Organizational Requirements 506
Hugo Estrada, Oscar Pastor, Alicia Martínez, Jose Torres-Jimenez

Using C++ Functors with Legacy C Libraries 514
Jan Broeckhove, Kurt Vanmechelen

Debugging of Java Programs Using HDT with Program Slicing 524
Hoon-Joon Kouh, Ki-Tae Kim, Sun-Moon Jo, Weon-Hee Yoo

Frameworks as Web Services 534
Olivia G. Fragoso Diaz, René Santaolaya Salgado, Isaac M. Vásquez Mendez, Manuel A. Valdés Marrero

Exception Rules Mining Based on Negative Association Rules 543
Olena Daly, David Taniar

A Reduced Codification for the Logical Representation of Job Shop Scheduling Problems 553
Juan Frausto-Solis, Marco Antonio Cruz-Chavez

Action Reasoning with Uncertain Resources 563
Alfredo Milani, Valentina Poggioni

Track on Security Engineering

Software Rejuvenation Approach to Security Engineering 574
Khin Mi Mi Aung, Jong Sou Park

A Rollback Recovery Algorithm for Intrusion Tolerant Intrusion Detection System 584
Myung-Kyu Yi, Chong-Sun Hwang

Design and Implementation of High-Performance Intrusion Detection System 594
Byoung-Koo Kim, Ik-Kyun Kim, Ki-Young Kim, Jong-Soo Jang

An Authenticated Key Agreement Protocol Resistant to a Dictionary Attack 603
Eun-Kyung Ryu, Kee-Won Kim, Kee-Young Yoo

A Study on Marking Bit Size for Path Identification Method: Deploying the Pi Filter at the End Host 611
Soon-Dong Kim, Man-Pyo Hong, Dong-Kyoo Kim

Efficient Password-Based Authenticated Key Agreement Protocol 617
Sung-Woon Lee, Woo-Hun Kim, Hyun-Sung Kim, Kee-Young Yoo

A Two-Public Key Scheme Omitting Collision Problem in Digital Signature 627
Sung Keun Song, Hee Yong Youn, Chang Won Park

A Novel Data Encryption and Distribution Approach for High Security and Availability Using LU Decomposition 637
Sung Jin Choi, Hee Yong Youn

An Efficient Conference Key Distribution System Based on Symmetric Balanced Incomplete Block Design 647
Youngjoo Cho, Changkyun Chi, Ilyong Chung

Multiparty Key Agreement Protocol with Cheater Identification Based on Shamir Secret Sharing 655
Kee-Young Yoo, Eun-Kyung Ryu, Jae-Yuel Im

Security of Shen et al.'s Timestamp-Based Password Authentication Scheme 665
Eun-Jun Yoon, Eun-Kyung Ryu, Kee-Young Yoo

ID-Based Authenticated Multiple-Key Agreement Protocol from Pairings 672
Kee-Won Kim, Eun-Kyung Ryu, Kee-Young Yoo

A Fine-Grained Taxonomy of Security Vulnerability in Active Network Environments 681
Jin S. Yang, Young J. Han, Dong S. Kim, Beom H. Chang, Tai M. Chung, Jung C. Na

A Secure and Flexible Multi-signcryption Scheme 689
Seung-Hyun Seo, Sang-Ho Lee

User Authentication Protocol Based on Human Memorable Password and Using RSA 698
IkSu Park, SeungBae Park, ByeongKyun Oh

Effective Packet Marking Approach to Defend against DDoS Attack 708
Heeran Lim, Manpyo Hong

A Relationship between Security Engineering and Security Evaluation . . . 717
Tai-hoon Kim, Haeng-kon Kim

A Relationship of Configuration Management Requirements between KISEC and ISO/IEC 15408 725
Hae-ki Lee, Jae-sun Shim, Seung Lee, Jong-bu Kim

Track on Information Systems and Information Technology

Term-Specific Language Modeling Approach to Text Categorization 735
Seung-Shik Kang

Context-Based Proofreading of Structured Documents 743
Won-Sung Sohn, Teuk-Seob Song, Jae-Kyung Kim, Yoon-Chul Choy, Kyong-Ho Lee, Sung-Bong Yang, Francis Neelamkavil

Implementation of New CTI Service Platform Using Voice XML 754
Jeong-Hoon Shin, Kwang-Seok Hong, Sung-Kyun Eom

Storing Together the Structural Information of XML Documents in Relational Databases 763
Min Jin, Byung-Joo Shin

Annotation Repositioning Methods in the XML Documents: Context-Based Approach 772
Won-Sung Sohn, Myeong-Cheol Ko, Hak-Keun Kim, Soon-Bum Lim, Yoon-Chul Choy

Isolating and Specifying the Relevant Information of an Organizational Model: A Process Oriented Towards Information System Generation 783
Alicia Martínez, Oscar Pastor, Hugo Estrada

A Weighted Fuzzy Min-Max Neural Network for Pattern Classification and Feature Extraction 791
Ho J. Kim, Tae W. Ryu, Thai T. Nguyen, Joon S. Lim, Sudhir Gupta

The eSAIDA Stream Authentication Scheme 799
Yongsu Park, Yookun Cho

An Object-Oriented Metric to Measure the Degree of Dependency Due to Unused Interfaces 808
René Santaolaya Salgado, Olivia G. Fragoso Diaz, Manuel A. Valdés Marrero, Isaac M. Vásquez Mendez, Sheila L. Delfín Lara

End-to-End QoS Management for VoIP Using DiffServ 818
Eun-Ju Ha, Byeong-Soo Yun

Multi-modal Biometrics System Using Face and Signature 828
Dae Jong Lee, Keun Chang Kwak, Jun Oh Min, Myung Geun Chun

Track on Information Retrieval

Using 3D Spatial Relationships for Image Retrieval by XML Annotation 838
SooCheol Lee, EenJun Hwang, YangKyoo Lee

Association Inlining for Mapping XML DTDs to Relational Tables 849
Byung-Joo Shin, Min Jin

XCRAB: A Content and Annotation-Based Multimedia Indexing and Retrieval System 859
SeungMin Rho, SooCheol Lee, EenJun Hwang, YangKyoo Lee

An Efficient Cache Conscious Multi-dimensional Index Structure 869
Jeong Min Shim, Seok Il Song, Young Soo Min, Jae Soo Yoo

Track on Image Processing

Tracking of Moving Objects Using Morphological Segmentation, Statistical Moments, and Radon Transform 877
Muhammad Bilal Ahmad, Min Hyuk Chang, Seung Jin Park, Jong An Park, Tae Sun Choi

Feature Extraction and Correlation for Time-to-Impact Segmentation Using Log-Polar Images 887
Fernando Pardo, Jose A. Boluda, Esther De Ves

Object Mark Segmentation Algorithm Using Dynamic Programming for Poor Quality Images in Automated Inspection Process 896
Dong-Joong Kang, Jong-Eun Ha, In-Mo Ahn

A Line-Based Pose Estimation Algorithm for 3-D Polyhedral Object Recognition 906
Tae-Jung Lho, Dong-Joong Kang, Jong-Eun Ha

Initialization Method for the Self-Calibration
Using Minimal Two Images 915
Jong-Eun Ha, Dong-Joong Kang

Face Recognition for Expressive Face Images 924
Hyoun-Joo Go, Keun Chang Kwak, Sung-Suk Kim, Myung-Geun Chun

Kolmogorov-Smirnov Test for Image Comparison 933
Eugene Demidenko

Modified Radius-Vector Function for Shape Contour Description 940
Sung Kwan Kang, Muhammad Bilal Ahmad, Jong Hun Chun, Pan Koo Kim, Jong An Park

Image Corner Detection Using Radon Transform 948
Seung Jin Park, Muhammad Bilal Ahmad, Rhee Seung-Hak, Seung Jo Han, Jong An Park

Analytical Comparison of Conventional and MCDF Operations
in Image Processing 956
Yinghua Lu, Wanwu Guo

On Extraction of Facial Features from Color Images 964
Jin Ok Kim, Jin Soo Kim, Young Ro Seo, Bum Ro Lee, Chin Hyun Chung, Key Seo Lee, Wha Young Yim, Sang Hyo Lee

Track on Networking

An Architecture for Mobility Management
in Mobile Computing Networks 974
Dohyeon Kim, Beongku An

An Adaptive Security Model for Heterogeneous Networks
Using MAUT and Simple Heuristics 983
Jongwoo Chae, Ghita Kouadri Mostéfaoui, Mokdong Chung

A Hybrid Restoration Scheme Based on Threshold Reaction Time
in Optical Burst-Switched Networks 994
Hae-Joung Lee, Kyu-Yeop Song, Won-Ho So, Jing Zhang, Debasish Datta, Biswanath Mukherjee, Young-Chon Kim

Author Index 1005

Table of Contents – Part I

Information Systems and Information Technologies (ISIT) Workshop, Multimedia Session

Face Detection by Facial Features with Color Images and Face Recognition Using PCA 1
Jin Ok Kim, Sung Jin Seo, Chin Hyun Chung, Jun Hwang, Woongjae Lee

A Shakable Snake for Estimation of Image Contours 9
Jin-Sung Yoon, Joo-Chul Park, Seok-Woo Jang, Gye-Young Kim

A New Recurrent Fuzzy Associative Memory for Recognizing Time-Series Patterns Contained Ambiguity 17
Joongjae Lee, Won Kim, Jeonghee Cha, Gyeyoung Kim, Hyungil Choi

A Novel Approach for Contents-Based E-catalogue Image Retrieval Based on a Differential Color Edge Model 25
Junchul Chun, Goorack Park, Changho An

A Feature-Based Algorithm for Recognizing Gestures on Portable Computers 33
Mi Gyung Cho, Am Sok Oh, Byung Kwan Lee

Fingerprint Matching Based on Linking Information Structure of Minutiae 41
JeongHee Cha, HyoJong Jang, GyeYoung Kim, HyungIl Choi

Video Summarization Using Fuzzy One-Class Support Vector Machine . . . 49
YoungSik Choi, KiJoo Kim

A Transcode and Prefetch Technique of Multimedia Presentations for Mobile Terminals 57
Maria Hong, Euisun Kang, Sungmin Um, Dongho Kim, Younghwan Lim

Information Systems and Information Technologies (ISIT) Workshop, Algorithm Session

A Study on Generating an Efficient Bottom-up Tree Rewrite Machine for JBurg 65
KyungWoo Kang

A Study on Methodology for Enhancing Reliability of Datapath 73
SunWoong Yang, MoonJoon Kim, JaeHeung Park, Hoon Chang

A Useful Method for Multiple Sequence Alignment and Its Implementation 81
Jin Kim, Dong-Hoi Kim, Saangyong Uhmn

A Research on the Stochastic Model for Spoken Language Understanding 89
Yong-Wan Roh, Kwang-Seok Hong, Hyon-Gu Lee

The Association Rule Algorithm with Missing Data in Data Mining 97
Bobby D. Gerardo, Jaewan Lee, Jungsik Lee, Mingi Park, Malrey Lee

Constructing Control Flow Graph for Java by Decoupling Exception Flow from Normal Flow 106
Jang-Wu Jo, Byeong-Mo Chang

On Negation-Based Conscious Agent 114
Kang Soo Tae, Hee Yong Youn, Gyung-Leen Park

A Document Classification Algorithm Using the Fuzzy Set Theory and Hierarchical Structure of Document 122
Seok-Woo Han, Hye-Jue Eun, Yong-Sung Kim, László T. Kóczy

A Supervised Korean Verb Sense Disambiguation Algorithm Based on Decision Lists of Syntactic Features 134
Kweon Yang Kim, Byong Gul Lee, Dong Kwon Hong

Information Systems and Information Technologies (ISIT) Workshop, Security Session

Network Security Management Using ARP Spoofing 142
Kyohyeok Kwon, Seongjin Ahn, Jin Wook Chung

A Secure and Practical CRT-Based RSA to Resist Side Channel Attacks 150
ChangKyun Kim, JaeCheol Ha, Sung-Hyun Kim, Seokyu Kim, Sung-Ming Yen, SangJae Moon

A Digital Watermarking Scheme in JPEG-2000 Using the Properties of Wavelet Coefficient Sign 159
Han-Ki Lee, Geun-Sil Song, Mi-Ae Kim, Kil-Sang Yoo, Won-Hyung Lee

A Security Proxy Based Protocol for Authenticating the Mobile IPv6 Binding Updates 167
Il-Sun You, Kyungsan Cho

A Fuzzy Expert System for Network Forensics 175
Jung-Sun Kim, Minsoo Kim, Bong-Nam Noh

A Design of Preventive Integrated Security Management System Using Security Labels and a Brief Comparison with Existing Models 183
D.S. Kim, T.M. Chung

The Vulnerability Assessment for Active Networks; Model, Policy, Procedures, and Performance Evaluations 191
Young J. Han, Jin S. Yang, Beom H. Chang, Jung C. Na, Tai M. Chung

Authentication of Mobile Node Using AAA in Coexistence of VPN and Mobile IP 199
Miyoung Kim, Misun Kim, Youngsong Mun

Survivality Modeling for Quantitative Security Assessment in Ubiquitous Computing Systems* 207
Changyeol Choi, Sungsoo Kim, We-Duke Cho

New Approach for Secure and Efficient Metering in the Web Advertising 215
Soon Seok Kim, Sung Kwon Kim, Hong Jin Park

MLS/SDM: Multi-level Secure Spatial Data Model 222
Young-Hwan Oh, Hae-Young Bae

Detection Techniques for ELF Executable File Using Assembly Instruction Searching 230
Jun-Hyung Park, Min-soo Kim, Bong-Nam Noh

Secure Communication Scheme Applying MX Resource Record in DNSSEC Domain 238
Hyung-Jin Lim, Hak-Ju Kim, Tae-Kyung Kim, Tai-Myung Chung

Committing Secure Results with Replicated Servers 246
Byoung Joon Min, Sung Ki Kim, Chaetae Im

Applied Research of Active Network to Control Network Traffic in Virtual Battlefield 254
Won Goo Lee, Jae Kwang Lee

Design and Implementation of the HoneyPot System with Focusing on the Session Redirection 262
Miyoung Kim, Misun Kim, Youngsong Mun

Information Systems and Information Technologies (ISIT) Workshop, Network Session

Analysis of Performance for MCVoD System 270
SeokHoon Kang, IkSoo Kim, Yoseop Woo

A QoS Improvement Scheme for Real-Time Traffic Using IPv6 Flow Labels 278
In Hwa Lee, Sung Jo Kim

Energy-Efficient Message Management Algorithms in HMIPv6 286
Sun Ok Yang, SungSuk Kim, Chong-Sun Hwang, SangKeun Lee

A Queue Management Scheme for Alleviating the Impact of Packet Size on the Achieved Throughput 294
Sungkeun Lee, Wongeun Oh, Myunghyun Song, Hyun Yoe, JinGwang Koh, Changryul Jung

PTrace: Pushback/SVM Based ICMP Traceback Mechanism against DDoS Attack 302
Hyung-Woo Lee, Min-Goo Kang, Chang-Won Choi

Traffic Control Scheme of ABR Service Using NLMS in ATM Network ... 310
Kwang-Ok Lee, Sang-Hyun Bae, Jin-Gwang Koh, Chang-Hee Kwon, Chong-Soo Cheung, In-Ho Ra

Information Systems and Information Technologies (ISIT) Workshop, Grid Session

XML-Based Workflow Description Language for Grid Applications 319
Yong-Won Kwon, So-Hyun Ryu, Chang-Sung Jeong, Hyoungwoo Park

Placement Algorithm of Web Server Replicas 328
Seonho Kim, Miyoun Yoon, Yongtae Shin

XML-OGL: UML-Based Graphical Web Query Language for XML Documents 337
Chang Yun Jeong, Yong-Sung Kim, Yan Ha

Layered Web-Caching Technique for VOD Services 345
Iksoo Kim, Yoseop Woo, Hyunchul Kang, Backhyun Kim, Jinsong Ouyang

QoS-Constrained Resource Allocation for a Grid-Based Multiple Source Electrocardiogram Application 352
Dong Su Nam, Chan-Hyun Youn, Bong Hwan Lee, Gari Clifford, Jennifer Healey

Efficient Pre-fetch and Pre-release Based Buffer Cache Management for Web Applications 360
Younghun Ko, Jaehyoun Kim, Hyunseung Choo

A New Architecture Design for Differentiated Resource Sharing on Grid Service 370
Eui-Nam Huh

An Experiment and Design of Web-Based Instruction Model for Collaboration Learning 378
Duckki Kim, Youngsong Mun

Information Systems and Information Technologies (ISIT) Workshop, Mobile Session

Performance Limitation of STBC OFDM-CDMA Systems in Mobile Fading Channels 386
Young-Hwan You, Tae-Won Jang, Min-Goo Kang, Hyung-Woo Lee, Hwa-Seop Lim, Yong-Soo Choi, Hyoung-Kyu Song

PMEPR Reduction Algorithms for STBC-OFDM Signals 394
Hyoung-Kyu Song, Min-Goo Kang, Ou-Seb Lee, Pan-Yuh Joo, We-Duke Cho, Mi-Jeong Kim, Young-Hwan You

An Efficient Image Transmission System Adopting OFDM Based Sequence Reordering Method in Non-flat Fading Channel 402
JaeMin Kwak, HeeGok Kang, SungEon Cho, Hyun Yoe, JinGwang Koh

The Efficient Web-Based Mobile GIS Service System through Reduction of Digital Map 410
Jong-Woo Kim, Seong-Seok Park, Chang-Soo Kim, Yugyung Lee

Reducing Link Loss in Ad Hoc Networks 418
Sangjoon Park, Eunjoo Jeong, Byunggi Kim

A Web Based Model for Analyzing Compliance of Mobile Content 426
Woojin Lee, Yongsun Cho, Kiwon Chong

Delay and Collision Reduction Mechanism for Distributed Fair Scheduling in Wireless LANs 434
Kee-Hyun Choi, Kyung-Soo Jang, Dong-Ryeol Shin

Approaches or Methods of Security Engineering Workshop

Bit-Serial Multipliers for Exponentiation and Division in $GF(2^m)$ Using Irreducible AOP 442
Yong Ho Hwang, Sang Gyoo Sim, Pil Joong Lee

Introduction and Evaluation of Development System Security Process of ISO/IEC TR 15504 451
Eun-ser Lee, Kyung Whan Lee, Tai-hoon Kim, Il-Hong Jung

Design on Mobile Secure Electronic Transaction Protocol with Component Based Development 461
Haeng-Kon Kim, Tai-Hoon Kim

A Distributed Online Certificate Status Protocol Based on GQ Signature Scheme 471
Dae Hyun Yum, Pil Joong Lee

A Design of Configuration Management Practices and CMPET in Common Criteria Based on Software Process Improvement Activity ... 481
Sun-Myung Hwang

The Design and Development for Risk Analysis Automatic Tool 491
Young-Hwan Bang, Yoon-Jung Jung, Injung Kim, Namhoon Lee, Gang-Soo Lee

A Fault-Tolerant Mobile Agent Model in Replicated Secure Services 500
Kyeongmo Park

Computation of Multiplicative Inverses in GF(2^n) Using Palindromic Representation 510
Hyeong Seon Yoo, Dongryeol Lee

A Study on Smart Card Security Evaluation Criteria for Side Channel Attacks 517
HoonJae Lee, ManKi Ahn, SeonGan Lim, SangJae Moon

User Authentication Protocol Based on Human Memorable Password and Using RSA 527
IkSu Park, SeungBae Park, ByeongKyun Oh

Supporting Adaptive Security Levels in Heterogeneous Environments 537
Ghita Kouadri Mostéfaoui, Mansoo Kim, Mokdong Chung

Intrusion Detection Using Noisy Training Data 547
Yongsu Park, Jaeheung Lee, Yookun Cho

A Study on Key Recovery Agent Protection Profile Having Composition Function 557
Dae-Hee Seo, Im-Yeong Lee, Hee-Un Park

Simulation-Based Security Testing for Continuity of Essential Service 567
Hyung-Jong Kim, JoonMo Kim, KangShin Lee, HongSub Lee, TaeHo Cho

NextPDM: Improving Productivity and Enhancing the Reusability with a Customizing Framework Toolkit 577
Ha Jin Hwang, Soung Won Kim

A Framework for Security Assurance in Component Based Development . 587
Hangkon Kim

An Information Engineering Methodology for the Security Strategy Planning 597
Sangkyun Kim, Choon Seong Leem

A Case Study in Applying Common Criteria to Development Process of Virtual Private Network 608
Sang ho Kim, Choon Seong Leem

A Pointer Forwarding Scheme for Fault-Tolerant Location Management in Mobile Networks 617
Ihn-Han Bae, Sun-Jin Oh

Architecture Environments for E-business Agent Based on Security 625
Ho-Jun Shin, Soo-Gi Lee

Authentication Authorization Accounting (AAA) Workshop

Multi-modal Biometrics System Using Face and Signature 635
Dae Jong Lee, Keun Chang Kwak, Jun Oh Min, Myung Geun Chun

Simple and Efficient Group Key Agreement Based on Factoring 645
Junghyun Nam, Seokhyang Cho, Seungjoo Kim, Dongho Won

On Facial Expression Recognition Using the Virtual Image Masking for a Security System 655
Jin Ok Kim, Kyong Sok Seo, Chin Hyun Chung, Jun Hwang, Woongjae Lee

Secure Handoff Based on Dual Session Keys in Mobile IP with AAA 663
Yumi Choi, Hyunseung Choo, Byong-Lyol Lee

Detection and Identification Mechanism against Spoofed Traffic Using Distributed Agents 673
Mihui Kim, Kijoon Chae

DMKB : A Defense Mechanism Knowledge Base 683
Eun-Jung Choi, Hyung-Jong Kim, Myuhng-Joo Kim

A Fine-Grained Taxonomy of Security Vulnerability in Active Network Environments 693
Jin S. Yang, Young J. Han, Dong S. Kim, Beom H. Chang, Tai M. Chung, Jung C. Na

A New Role-Based Authorization Model
in a Corporate Workflow Systems 701
HyungHyo Lee, SeungYong Lee, Bong-Nam Noh

A New Synchronization Protocol for Authentication
in Wireless LAN Environment 711
Hea Suk Jo, Hee Yong Youn

A Robust Image Authentication Method Surviving Acceptable
Modifications 722
Mi-Ae Kim, Geun-Sil Song, Won-Hyung Lee

Practical Digital Signature Generation Using Biometrics 728
Taekyoung Kwon, Jae-il Lee

Performance Improvement in Mobile IPv6 Using AAA
and Fast Handoff 738
Changnam Kim, Young-Sin Kim, Eui-Nam Huh, Youngsong Mun

An Efficient Key Agreement Protocol for Secure Authentication 746
Young-Sin Kim, Eui-Nam Huh, Jun Hwang, Byung-Wook Lee

A Policy-Based Security Management Architecture Using XML
Encryption Mechanism for Improving SNMPv3 755
Choong Seon Hong, Joon Heo

IDentification Key Based AAA Mechanism in Mobile IP Networks 765
Hoseong Jeon, Hyunseung Choo, Jai-Ho Oh

An Integrated XML Security Mechanism for Mobile Grid Application 776
Kiyoung Moon, Namje Park, Jongsu Jang, Sungwon Sohn, Jaecheol Ryou

Development of XKMS-Based Service Component for Using PKI
in XML Web Services Environment 784
Namje Park, Kiyoung Moon, Jongsu Jang, Sungwon Sohn

A Scheme for Improving WEP Key Transmission between APs
in Wireless Environment 792
Chi Hyung In, Choong Seon Hong, Il Gyu Song

Internet Communication Security Workshop

Generic Construction of Certificateless Encryption 802
Dae Hyun Yum, Pil Joong Lee

Security Issues in Network File Systems 812
Antonio Izquierdo, Jose María Sierra, Julio César Hernández, Arturo Ribagorda

A Content-Independent Scalable Encryption Model 821
Stefan Lindskog, Johan Strandbergh, Mikael Hackman, Erland Jonsson

Fair Exchange to Achieve Atomicity in Payments of High Amounts Using Electronic Cash 831
Magdalena Payeras-Capella, Josep Lluís Ferrer-Gomila, Llorenç Huguet-Rotger

N3: A Geometrical Approach for Network Intrusion Detection at the Application Layer 841
Juan M. Estévez-Tapiador, Pedro García-Teodoro, Jesús E. Díaz-Verdejo

Validating the Use of BAN LOGIC 851
José María Sierra, Julio César Hernández, Almudena Alcaide, Joaquín Torres

Use of Spectral Techniques in the Design of Symmetrical Cryptosystems 859
Luis Javier García Villalba

Load Balancing and Survivability for Network Services Based on Intelligent Agents 868
Robson de Oliveira Albuquerque, Rafael T. de Sousa Jr., Tamer Américo da Silva, Ricardo S. Puttini, Clàudia Jacy Barenco Abbas, Luis Javier García Villalba

A Scalable PKI for Secure Routing in the Internet 882
Francesco Palmieri

Cryptanalysis and Improvement of Password Authenticated Key Exchange Scheme between Clients with Different Passwords 895
Jeeyeon Kim, Seungjoo Kim, Jin Kwak, Dongho Won

Timeout Estimation Using a Simulation Model for Non-repudiation Protocols 903
Mildrey Carbonell, Jose A. Onieva, Javier Lopez, Deborah Galpert, Jianying Zhou

DDoS Attack Defense Architecture Using Active Network Technology 915
Choong Seon Hong, Yoshiaki Kasahara, Dea Hwan Lee

A Voting System with Trusted Verifiable Services 924
Macià Mut Puigserver, Josep Lluís Ferrer Gomila, Llorenç Huguet i Rotger

Chaotic Protocols 938
Mohamed Mejri

Security Consequences of Messaging Hubs in Many-to-Many E-procurement Solutions 949
Eva Ponce, Alfonso Durán, Teresa Sánchez

The SAC Test: A New Randomness Test, with Some Applications to PRNG Analysis 960
Julio César Hernandez, José María Sierra, Andre Seznec

A Survey of Web Services Security 968
Carlos Gutiérrez, Eduardo Fernández-Medina, Mario Piattini

Fair Certified E-mail Protocols with Delivery Deadline Agreement 978
Yongsu Park, Yookun Cho

Location Management and the Security in the Next Generation Mobile Networks Workshop

QS-Ware: The Middleware for Providing QoS and Secure Ability to Web Server 988
Seung-won Shin, Kwang-ho Baik, Ki-Young Kim, Jong-Soo Jang

Implementation and Performance Evaluation of High-Performance Intrusion Detection and Response System 998
Hyeong-Ju Kim, Byoung-Koo Kim, Ik-Kyun Kim

Efficient Key Distribution Protocol for Secure Multicast Communication 1007
Bonghan Kim, Hanjin Cho, Jae Kwang Lee

A Bayesian Approach for Estimating Link Travel Time on Urban Arterial Road Network 1017
Taehyung Park, Sangkeon Lee

Perimeter Defence Policy Model of Cascade MPLS VPN Networks 1026
Won Shik Na, Jeom Goo Kim, Intae Ryoo

Design of Authentication and Key Exchange Protocol in Ethernet Passive Optical Networks 1035
Sun-Sik Roh, Su-Hyun Kim, Gwang-Hyun Kim

Detection of Moving Objects Edges to Implement Home Security System in a Wireless Environment 1044
Yonghak Ahn, Kiok Ahn, Oksam Chae

Reduction Method of Threat Phrases by Classifying Assets 1052
Tai-Hoon Kim, Dong Chun Lee

Anomaly Detection Using Sequential Properties of Packets in Mobile Environment 1060
Seong-sik Hong, Hwang-bin Ryou

A Case Study in Applying Common Criteria to Development Process to Improve Security of Software Products 1069
Sang Ho Kim, Choon Seong Leem

A New Recovery Scheme with Reverse Shared Risk Link Group in GMPLS-Based WDM Networks 1078
Hyuncheol Kim, Seongjin Ahn, Daeho Kim, Sunghae Kim, Jin Wook Chung

Real Time Estimation of Bus Arrival Time under Mobile Environment ... 1088
Taehyung Park, Sangkeon Lee, Young-Jun Moon

Call Tracking and Location Updating Using DHS in Mobile Networks ... 1097
Dong Chun Lee

Routing and Handoff Workshop

Improving TCP Performance over Mobile IPv6 1105
Young-Chul Shim, Nam-Chang Kim, Ho-Seok Kang

Design of Mobile Network Route Optimization Based on the Hierarchical Algorithm 1115
Dongkeun Lee, Keecheon Kim, Sunyoung Han

On Algorithms for Minimum-Cost Quickest Paths with Multiple Delay-Bounds 1125
Young-Cheol Bang, Inki Hong, Sungchang Lee, Byungjun Ahn

A Fast Handover Protocol for Mobile IPv6 Using Mobility Prediction Mechanism 1134
Dae Sun Kim, Choong Seon Hong

The Layer 2 Handoff Scheme for Mobile IP over IEEE 802.11 Wireless LAN 1144
Jongjin Park, Youngsong Mun

Session Key Exchange Based on Dynamic Security Association for Mobile IP Fast Handoff 1151
Hyun Gon Kim, Doo Ho Choi

A Modified AODV Protocol with Multi-paths Considering Classes of Services 1159
Min-Su Kim, Ki Jin Kwon, Min Young Chung, Tae-Jin Lee, Jaehyung Park

Author Index 1169

Table of Contents – Part II

Grid Computing Workshop

Advanced Simulation Technique for Modeling Multiphase Fluid Flow in Porous Media 1
Jong G. Kim, Hyoung Woo Park

The P-GRADE Grid Portal 10
Csaba Németh, Gábor Dózsa, Róbert Lovas, Péter Kacsuk

A Smart Agent-Based Grid Computing Platform 20
Kwang-Won Koh, Hie-Cheol Kim, Kyung-Lang Park, Hwang-Jik Lee, Shin-Dug Kim

Publishing and Executing Parallel Legacy Code Using an OGSI Grid Service 30
T. Delaitre, A. Goyeneche, T. Kiss, S.C. Winter

The PROVE Trace Visualisation Tool as a Grid Service 37
Gergely Sipos, Péter Kacsuk

Privacy Protection in Ubiquitous Computing Based on Privacy Label and Information Flow 46
Seong Oun Hwang, Ki Song Yoon

Resource Management and Scheduling Techniques for Cluster and Grid Computing Systems Workshop

Application-Oriented Scheduling in the Knowledge Grid: A Model and Architecture 55
Andrea Pugliese, Domenico Talia

A Monitoring and Prediction Tool for Time-Constraint Grid Application 66
Abdulla Othman, Karim Djemame, Iain Gourlay

Optimal Server Allocation in Reconfigurable Clusters with Multiple Job Types 76
J. Palmer, I. Mitrani

Design and Evaluation of an Agent-Based Communication Model for a Parallel File System 87
María S. Pérez, Alberto Sánchez, Jemal Abawajy, Víctor Robles, José M. Peña

Task Allocation for Minimizing Programs Completion Time in Multicomputer Systems 97
Gamal Attiya, Yskandar Hamam

Fault Detection Service Architecture for Grid Computing Systems 107
J.H. Abawajy

Adaptive Interval-Based Caching Management Scheme for Cluster Video Server 116
Qin Zhang, Hai Jin, Yufu Li, Shengli Li

A Scalable Streaming Proxy Server Based on Cluster Architecture 126
Hai Jin, Jie Chu, Kaiqin Fan, Zhi Dong, Zhiling Yang

The Measurement of an Optimum Load Balancing Algorithm in a Master/Slave Architecture 136
Finbarr O'Loughlin, Desmond Chambers

Data Discovery Mechanism for a Large Peer-to-Peer Based Scientific Data Grid Environment 146
Azizol Abdullah, Mohamed Othman, Md Nasir Sulaiman, Hamidah Ibrahim, Abu Talib Othman

A DAG-Based XCIGS Algorithm for Dependent Tasks in Grid Environments 158
Changqin Huang, Deren Chen, Qinghuai Zeng, Hualiang Hu

Running Data Mining Applications on the Grid: A Bag-of-Tasks Approach 168
Fabrício A.B. da Silva, Sílvia Carvalho, Hermes Senger, Eduardo R. Hruschka, Cléver R.G. de Farias

Parallel and Distributed Computing Workshop

Application of Block Design to a Load Balancing Algorithm on Distributed Networks 178
Yeijin Lee, Okbin Lee, Taehoon Lee, Ilyong Chung

Maintenance Strategy for Efficient Communication at Data Warehouse .. 186
Hyun Chang Lee, Sang Hyun Bae

Conflict Resolution of Data Synchronization in Mobile Environment 196
YoungSeok Lee, YounSoo Kim, Hoon Choi

A Framework for Orthogonal Data and Control Parallelism Exploitation 206
S. Campa, M. Danelutto

Multiplier with Parallel CSA Using CRT's Specific Moduli $(2^k\text{-}1, 2^k, 2^k\text{+}1)$ 216
Wu Woan Kim, Sang-Dong Jang

Unified Development Solution for Cluster and Grid Computing and Its Application in Chemistry 226
Róbert Lovas, Péter Kacsuk, István Lagzi, Tamás Turányi

Remote Visualization Based on Grid Computing 236
Zhigeng Pan, Bailin Yang, Mingmin Zhang, Qizhi Yu, Hai Lin

Avenues for High Performance Computation on a PC 246
Yu-Fai Fung, M. Fikret Ercan, Wai-Leung Cheung, Gujit Singh

A Modified Parallel Computation Model Based on Cluster 252
Xiaotu Li, Jizhou Sun, Jiawan Zhang, Zhaohui Qi, Gang Li

Parallel Testing Method by Partitioning Circuit Based on the Exhaustive Test 262
Wu Woan Kim

A Parallel Volume Splatting Algorithm Based on PC-Clusters 272
Jiawan Zhang, Jizhou Sun, Yi Zhang, Qianqian Han, Zhou Jin

Molecular Processes Simulation Workshop

Three-Center Nuclear Attraction Integrals for Density Functional Theory and Nonlinear Transformations 280
Hassan Safouhi

Parallelization of Reaction Dynamics Codes Using P-GRADE: A Case Study 290
Ákos Bencsura, György Lendvay

Numerical Implementation of Quantum Fluid Dynamics: A Working Example 300
Fabrizio Esposito

Numerical Revelation and Analysis of Critical Ignition Conditions for Branch Chain Reactions by Hamiltonian Systematization Methods of Kinetic Models 313
Gagik A. Martoyan, Levon A. Tavadyan

Computer Simulations in Ion-Atom Collisions 321
S.F.C. O'Rourke, R.T. Pedlow, D.S.F. Crothers

Bond Order Potentials for a priori Simulations of Polyatomic Reactions 328
Ernesto Garcia, Carlos Sánchez, Margarita Albertí, Antonio Laganà

Inorganic Phosphates Investigation by Support Vector Machine 338
Cinzia Pierro, Francesco Capitelli

Characterization of Equilibrium Structure for N_2-N_2 Dimer in 1.2Å≤R≥2.5Å Region Using DFT Method 350
Ajmal H. Hamdani, S. Shahdin

A Time Dependent Study of the Nitrogen Atom Nitrogen Molecule Reaction 357
Antonio Laganà, Leonardo Pacifici, Dimitris Skouteris

From DFT Cluster Calculations to Molecular Dynamics Simulation of N_2 Formation on a Silica Model Surface 366
M. Cacciatore, A. Pieretti, M. Rutigliano, N. Sanna

Molecular Mechanics and Dynamics Calculations to Bridge Molecular Structure Information and Spectroscopic Measurements on Complexes of Aromatic Compounds 374
G. Pietraperzia, R. Chelli, M. Becucci, Antonio Riganelli, Margarita Alberti, Antonio Laganà

Direct Simulation Monte Carlo Modeling of Non Equilibrium Reacting Flows. Issues for the Inclusion into a ab initio Molecular Processes Simulator 383
D. Bruno, M. Capitelli, S. Longo, P. Minelli

Molecular Simulation of Reaction and Adsorption in Nanochemical Devices: Increase of Reaction Conversion by Separation of a Product from the Reaction Mixture 392
William R. Smith, Martin Lísal

Quantum Generalization of Molecular Dynamics Method. Wigner Approach 402
V. Filinov, M. Bonitz, V. Fortov, P. Levashov

$C_6NH_6^+$ Ions as Intermediates in the Reaction between Benzene and N^+ Ions 412
Marco Di Stefano, Marzio Rosi, Antonio Sgamellotti

Towards a Full Dimensional Exact Quantum Calculation of the Li + HF Reactive Cross Section 422
Antonio Laganà, Stefano Crocchianti, Valentina Piermarini

Conformations of 1,2,4,6-Tetrathiepane 432
Issa Yavari, Arash Jabbari, Shahram Moradi

Fine Grain Parallelization of a Discrete Variable Wavepacket Calculation Using ASSIST-CL 437
Stefano Gregori, Sergio Tasso, Antonio Laganà

Numerical Models in Biomechanics Session

On the Solution of Contact Problems with Visco-Plastic Friction in the Bingham Rheology: An Application in Biomechanics 445
Jiří Nedoma

On the Stress-Strain Analysis of the Knee Replacement 456
J. Daněk, F. Denk, I. Hlaváček, Jiří Nedoma, J. Stehlík, P. Vavřík

Musculoskeletal Modeling of Lumbar Spine under Follower Loads 467
Yoon Hyuk Kim, Kyungsoo Kim

Computational Approach to Optimal Transport Network Construction in Biomechanics 476
Natalya Kizilova

Encoding Image Based on Retinal Ganglion Cell 486
Sung-Kwan Je, Eui-Young Cha, Jae-Hyun Cho

Scientific Computing Environments (SCE's) for Imaging in Science Session

A Simple Data Analysis Method for Kinetic Parameters Estimation from Renal Measurements with a Three-Headed SPECT System 495
Eleonora Vanzi, Andreas Robert Formiconi

Integrating Medical Imaging into a Grid Based Computing Infrastructure 505
Paola Bonetto, Mario Guarracino, Fabrizio Inguglia

Integrating Scientific Software Libraries in Problem Solving Environments: A Case Study with ScaLAPACK 515
L. D'Amore, Mario R. Guarracino, G. Laccetti, A. Murli

Parallel/Distributed Film Line Scratch Restoration by Fusion Techniques 525
G. Laccetti, L. Maddalena, A. Petrosino

An Interactive Distributed Environment for Digital Film Restoration 536
F. Collura, A. Machì, F. Nicotra

Computer Graphics and Geometric Modeling Workshop (TSCG 2004)

On Triangulations 544
Ivana Kolingerová

Probability Distribution of Op-Codes in Edgebreaker 554
Deok-Soo Kim, Cheol-Hyung Cho, Youngsong Cho, Chang Wook Kang, Hyun Chan Lee, Joon Young Park

Polyhedron Splitting Algorithm for 3D Layer Generation 564
Jaeho Lee, Joon Young Park, Deok-Soo Kim, Hyun Chan Lee

Synthesis of Mechanical Structures Using a Genetic Algorithm 573
In-Ho Lee, Joo-Heon Cha, Jay-Jung Kim, M.-W. Park

Optimal Direction for Monotone Chain Decomposition 583
Hayong Shin, Deok-Soo Kim

GTVIS: Fast and Efficient Rendering System for Real-Time Terrain Visualization 592
Russel A. Apu, Marina L. Gavrilova

Target Data Projection in Multivariate Visualization – An Application to Mine Planning 603
Leonardo Soto, Ricardo Sánchez, Jorge Amaya

Parametric Freehand Sketches 613
Ferran Naya, Manuel Contero, Nuria Aleixos, Joaquim Jorge

Variable Level of Detail Strips 622
J.F. Ramos, M. Chover

Bézier Solutions of the Wave Equation 631
J.V. Beltran, J. Monterde

Matlab Toolbox for a First Computer Graphics Course for Engineers 641
Akemi Gálvez, A. Iglesias, César Otero, Reinaldo Togores

A Differential Method for Parametric Surface Intersection 651
A. Gálvez, J. Puig-Pey, A. Iglesias

A Comparison Study of Metaheuristic Techniques for Providing QoS to Avatars in DVE Systems 661
P. Morillo, J.M. Orduña, Marcos Fernández, J. Duato

Visualization of Large Terrain Using Non-restricted Quadtree Triangulations 671
Mariano Pérez, Ricardo Olanda, Marcos Fernández

Boundary Filtering in Surface Reconstruction 682
Michal Varnuška, Ivana Kolingerová

Image Coherence Based Adaptive Sampling for Image Synthesis 693
Qing Xu, Roberto Brunelli, Stefano Messelodi, Jiawan Zhang, Mingchu Li

A Comparison of Multiresolution Modelling in Real-Time Terrain Visualisation 703
C. Rebollo, I. Remolar, M. Chover, J.F. Ramos

Photo-realistic 3D Head Modeling Using Multi-view Images 713
Tong-Yee Lee, Ping-Hsien Lin, Tz-Hsien Yang

Texture Mapping on Arbitrary 3D Surfaces 721
Tong-Yee Lee, Shaur-Uei Yan

Segmentation-Based Interpolation of 3D Medical Images 731
Zhigeng Pan, Xuesong Yin, Guohua Wu

A Bandwidth Reduction Scheme for 3D Texture-Based Volume Rendering on Commodity Graphics Hardware 741
Won-Jong Lee, Woo-Chan Park, Jung-Woo Kim, Tack-Don Han, Sung-Bong Yang, Francis Neelamkavil

An Efficient Image-Based 3D Reconstruction Algorithm for Plants 751
Zhigeng Pan, Weixi Hu, Xinyu Guo, Chunjiang Zhao

Where the Truth Lies (in Automatic Theorem Proving in Elementary Geometry) 761
T. Recio, F. Botana

Helical Curves on Surfaces for Computer-Aided Geometric Design and Manufacturing 771
J. Puig-Pey, Akemi Gálvez, A. Iglesias

An Application of Computer Graphics for Landscape Impact Assessment 779
César Otero, Viola Bruschi, Antonio Cendrero, Akemi Gálvez, Miguel Lázaro, Reinaldo Togores

Fast Stereo Matching Using Block Similarity 789
Han-Suh Koo, Chang-Sung Jeong

View Morphing Based on Auto-calibration for Generation of In-between Views 799
Jin-Young Song, Yong-Ho Hwang, Hyun-Ki Hong

Virtual Reality in Scientific Applications and Learning (VRSAL 2004) Workshop

Immersive Displays Based on a Multi-channel PC Clustered System 809
Hunjoo Lee, Kijong Byun

Virtual Reality Technology Applied to Simulate Construction Processes 817
Alcínia Zita Sampaio, Pedro Gameiro Henriques, Pedro Studer

Virtual Reality Applied to Molecular Sciences 827
Osvaldo Gervasi, Antonio Riganelli, Antonio Laganà

Design and Implementation of an Online 3D Game Engine 837
Hunjoo Lee, Taejoon Park

Dynamically Changing Road Networks – Modelling and Visualization in Real Time 843
Christian Mark, Armin Kaußner, Martin Grein, Hartmut Noltemeier

EoL: A Web-Based Distance Assessment System 854
Osvaldo Gervasi, Antonio Laganà

Discovery Knowledge of User Preferences: Ontologies in Fashion Design Recommender Agent System 863
Kyung-Yong Jung, Young-Joo Na, Dong-Hyun Park, Jung-Hyun Lee

When an Ivy League University Puts Its Courses Online, Who's Going to Need a Local University? 873
Matthew C.F. Lau, Rebecca B.N. Tan

Web-Based Learning Session

Threads in an Undergraduate Course: A Java Example Illuminating Different Multithreading Approaches 882
H. Martin Bücker, Bruno Lang, Hans-Joachim Pflug, Andre Vehreschild

A Comparison of Web Searching Strategies According to Cognitive Styles of Elementary Students 892
Hanil Kim, Miso Yun, Pankoo Kim

The Development and Application of a Web-Based Information Communication Ethics Education System 902
Suk-Ki Hong, Woochun Jun

An Interaction Model for Web-Based Learning: Cooperative Project 913
Eunhee Choi, Woochun Jun, Suk-Ki Hong, Young-Cheol Bang

Observing Standards for Web-Based Learning from the Web 922
Luis Anido, Judith Rodríguez, Manuel Caeiro, Juan Santos

Matrix Approximations with Applications to Science, Engineering, and Computer Science Workshop

On Computing the Spectral Decomposition of Symmetric Arrowhead Matrices 932
Fasma Diele, Nicola Mastronardi, Marc Van Barel, Ellen Van Camp

Relevance Feedback for Content-Based Image Retrieval Using Proximal Support Vector Machine 942
YoungSik Choi, JiSung Noh

Orthonormality-Constrained INDSCAL with Nonnegative Saliences 952
Nickolay T. Trendafilov

Optical Flow Estimation via Neural Singular Value Decomposition Learning 961
Simone Fiori, Nicoletta Del Buono, Tiziano Politi

Numerical Methods Based on Gaussian Quadrature and Continuous Runge-Kutta Integration for Optimal Control Problems 971
Fasma Diele, Carmela Marangi, Stefania Ragni

Graph Adjacency Matrix Associated with a Data Partition 979
Giuseppe Acciani, Girolamo Fornarelli, Luciano Liturri

A Continuous Technique for the Weighted Low-Rank Approximation Problem 988
Nicoletta Del Buono, Tiziano Politi

Spatial Statistics and Geographical Information Systems: Algorithms and Applications

A Spatial Multivariate Approach to the Analysis of Accessibility to Health Care Facilities in Canada 998
Stefania Bertazzon

Density Analysis on Large Geographical Databases. Search for an Index of Centrality of Services at Urban Scale 1009
Giuseppe Borruso, Gabriella Schoier

An Exploratory Spatial Data Analysis (ESDA) Toolkit for the Analysis of Activity/Travel Data 1016
Ronald N. Buliung, Pavlos S. Kanaroglou

Using Formal Ontology for Integrated Spatial Data Mining 1026
Sungsoon Hwang

G.I.S. and Fuzzy Sets for the Land Suitability Analysis 1036
Beniamino Murgante, Giuseppe Las Casas

Intelligent Gis and Retail Location Dynamics: A Multi Agent System Integrated with ArcGis 1046
S. Lombardo, M. Petri, D. Zotta

ArcObjects Development in Zone Design Using Visual Basic for Applications 1057
Sergio Palladini

Searching for 2D Spatial Network Holes 1069
Femke Reitsma, Shane Engel

Extension of Geography Markup Language (GML) for Mobile and Location-Based Applications 1079
Young Soo Ahn, Soon-Young Park, Sang Bong Yoo, Hae-Young Bae

A Clustering Method for Large Spatial Databases 1089
Gabriella Schoier, Giuseppe Borruso

GeoSurveillance: Software for Monitoring Change in Geographic Patterns 1096
Peter Rogerson, Ikuho Yamada

From Axial Maps to Mark Point Parameter Analysis (Ma.P.P.A.) – A GIS Implemented Method to Automate Configurational Analysis 1107
V. Cutini, M. Petri, A. Santucci

Computing Foraging Paths for Shore-Birds Using Fractal Dimensions and Pecking Success from Footprint Surveys on Mudflats: An Application for Red-Necked Stints in the Moroshechnaya River Estuary, Kamchatka-Russian Far East 1117
Falk Huettmann

Author Index 1129

Table of Contents – Part III

Workshop on Computational Geometry and Applications (CGA 04)

Geometric Graphs Realization as Coin Graphs 1
Manuel Abellanas, Carlos Moreno-Jiménez

Disc Covering Problem with Application to Digital Halftoning 11
Tetsuo Asano, Peter Brass, Shinji Sasahara

On Local Transformations in Plane Geometric Graphs Embedded on Small Grids 22
Manuel Abellanas, Prosenjit Bose, Alfredo García, Ferran Hurtado, Pedro Ramos, Eduardo Rivera-Campo, Javier Tejel

Reducing the Time Complexity of Minkowski-Sum Based Similarity Calculations by Using Geometric Inequalities 32
Henk Bekker, Axel Brink

A Practical Algorithm for Approximating Shortest Weighted Path between a Pair of Points on Polyhedral Surface 42
Sasanka Roy, Sandip Das, Subhas C. Nandy

Plane-Sweep Algorithm of O(nlogn) for the Inclusion Hierarchy among Circles 53
Deok-Soo Kim, Byunghoon Lee, Cheol-Hyung Cho, Kokichi Sugihara

Shortest Paths for Disc Obstacles 62
Deok-Soo Kim, Kwangseok Yu, Youngsong Cho, Donguk Kim, Chee Yap

Improving the Global Continuity of the Natural Neighbor Interpolation 71
Hisamoto Hiyoshi, Kokichi Sugihara

Combinatories and Triangulations 81
Tomas Hlavaty, Václav Skala

Approximations for Two Decomposition-Based Geometric Optimization Problems 90
Minghui Jiang, Brendan Mumey, Zhongping Qin, Andrew Tomascak, Binhai Zhu

Computing Largest Empty Slabs 99
Jose Miguel Díaz-Báñez, Mario Alberto López, Joan Antoni Sellarès

3D-Color-Structure-Code – A New Non-plainness Island Hierarchy 109
Patrick Sturm

Quadratic-Time Linear-Space Algorithms for Generating Orthogonal Polygons with a Given Number of Vertices 117
Ana Paula Tomás, António Leslie Bajuelos

Partitioning Orthogonal Polygons by Extension of All Edges Incident to Reflex Vertices: Lower and Upper Bounds on the Number of Pieces 127
António Leslie Bajuelos, Ana Paula Tomás, Fábio Marques

On the Time Complexity of Rectangular Covering Problems in the Discrete Plane 137
Stefan Porschen

Approximating Smallest Enclosing Balls 147
Frank Nielsen, Richard Nock

Geometry Applied to Designing Spatial Structures: Joining Two Worlds 158
José Andrés Díaz, Reinaldo Togores, César Otero

A Robust and Fast Algorithm for Computing Exact and Approximate Shortest Visiting Routes 168
Håkan Jonsson

Automated Model Generation System Based on Freeform Deformation and Genetic Algorithm 178
Hyunpung Park, Kwan H. Lee

Speculative Parallelization of a Randomized Incremental Convex Hull Algorithm 188
Marcelo Cintra, Diego R. Llanos, Belén Palop

The Employment of Regular Triangulation for Constrained Delaunay Triangulation 198
Pavel Maur, Ivana Kolingerová

The Anchored Voronoi Diagram 207
Jose Miguel Díaz-Báñez, Francisco Gómez, Immaculada Ventura

Implementation of the Voronoi-Delaunay Method for Analysis of Intermolecular Voids 217
A.V. Anikeenko, M.G. Alinchenko, V.P. Voloshin, N.N. Medvedev, M.L. Gavrilova, P. Jedlovszky

Approximation of the Boat-Sail Voronoi Diagram and Its Application 227
Tetsushi Nishida, Kokichi Sugihara

Incremental Adaptive Loop Subdivision 237
Hamid-Reza Pakdel, Faramarz F. Samavati

Reverse Subdivision Multiresolution for Polygonal Silhouette Error Correction 247
Kevin Foster, Mario Costa Sousa, Faramarz F. Samavati, Brian Wyvill

Cylindrical Approximation of a Neuron from Reconstructed Polyhedron 257
Wenhao Lin, Binhai Zhu, Gwen Jacobs, Gary Orser

Skeletizing 3D-Objects by Projections 267
David Ménegaux, Dominique Faudot, Hamamache Kheddouci

Track on Computational Geometry

An Efficient Algorithm for Determining 3-D Bi-plane Imaging Geometry 277
Jinhui Xu, Guang Xu, Zhenming Chen, Kenneth R. Hoffmann

Error Concealment Method Using Three-Dimensional Motion Estimation 288
Dong-Hwan Choi, Sang-Hak Lee, Chan-Sik Hwang

Confidence Sets for the Aumann Mean of a Random Closed Set 298
Raffaello Seri, Christine Choirat

An Algorithm of Mapping Additional Scalar Value in 2D Vector Field Visualization 308
Zhigeng Pan, Jianfeng Lu, Minming Zhang

Network Probabilistic Connectivity: Exact Calculation with Use of Chains 315
Olga K. Rodionova, Alexey S. Rodionov, Hyunseung Choo

Curvature Dependent Polygonization by the Edge Spinning 325
Martin Čermák, Václav Skala

SOM: A Novel Model for Defining Topological Line-Region Relations 335
Xiaolin Wang, Yingwei Luo, Zhuoqun Xu

Track on Adaptive Algorithms

On Automatic Global Error Control in Multistep Methods with Polynomial Interpolation of Numerical Solution 345
Gennady Yu. Kulikov, Sergey K. Shindin

Approximation Algorithms for k-Source Bottleneck Routing Cost Spanning Tree Problems 355
Yen Hung Chen, Bang Ye Wu, Chuan Yi Tang

Efficient Sequential and Parallel Algorithms for Popularity Computation on the World Wide Web with Applications against Spamming 367
Sung-Ryul Kim

Decentralized Inter-agent Message Forwarding Protocols for Mobile Agent Systems 376
JinHo Ahn

Optimization of Usability on an Authentication System Built from Voice and Neural Networks 386
Tae-Seung Lee, Byong-Won Hwang

An Efficient Simple Cooling Schedule for Simulated Annealing 396
Mir M. Atiqullah

A Problem-Specific Convergence Bound for Simulated Annealing-Based Local Search 405
Andreas A. Albrecht

Comparison and Selection of Exact and Heuristic Algorithms 415
Joaquín Pérez O., Rodolfo A. Pazos R., Juan Frausto-Solís, Guillermo Rodríguez O., Laura Cruz R., Héctor Fraire H.

Adaptive Texture Recognition in Image Sequences with Prediction through Features Interpolation 425
Sung Baik, Ran Baik

Fuzzy Matching of User Profiles for a Banner Engine 433
Alfredo Milani, Chiara Morici, Radoslaw Niewiadomski

Track on Biology, Biochemistry, Bioinformatics

Genome Database Integration 443
Andrew Robinson, Wenny Rahayu

Protein Structure Prediction with Stochastic Optimization Methods: Folding and Misfolding the Villin Headpiece 454
Thomas Herges, Alexander Schug, Wolfgang Wenzel

High Throughput *in-silico* Screening against Flexible Protein Receptors 465
Holger Merlitz, Wolfgang Wenzel

A Sequence-Focused Parallelisation of EMBOSS on a Cluster of Workstations 473
Karl Podesta, Martin Crane, Heather J. Ruskin

A Parallel Solution to Reverse Engineering Genetic Networks 481
Dorothy Bollman, Edusmildo Orozco, Oscar Moreno

Deformable Templates for Recognizing the Shape of the Zebra Fish Egg Cell 489
Ho-Dong Lee, Min-Soo Jang, Seok-Joo Lee, Yong-Guk Kim, Byungkyu Kim, Gwi-Tae Park

Multiple Parameterisation of Human Immune Response in HIV: Many-Cell Models 498
Yu Feng, Heather J. Ruskin, Yongle Liu

Track on Cluster Computing

Semantic Completeness in Sub-ontology Extraction Using Distributed Methods 508
Mehul Bhatt, Carlo Wouters, Andrew Flahive, Wenny Rahayu, David Taniar

Distributed Mutual Exclusion Algorithms on a Ring of Clusters 518
Kayhan Erciyes

A Cluster Based Hierarchical Routing Protocol for Mobile Networks 528
Kayhan Erciyes, Geoffrey Marshall

Distributed Optimization of Fiber Optic Network Layout Using MATLAB 538
Roman Pfarrhofer, Markus Kelz, Peter Bachhiesl, Herbert Stögner, Andreas Uhl

Cache Conscious Dynamic Transaction Routing in a Shared Disks Cluster 548
Kyungoh Ohn, Haengrae Cho

A Personalized Recommendation Agent System for E-mail Document Classification 558
Ok-Ran Jeong, Dong-Sub Cho

An Adaptive Prefetching Method for Web Caches 566
Jaeeun Jeon, Gunhoon Lee, Ki Dong Lee, Byoungchul Ahn

Track on Computational Medicine

Image Processing and Retinopathy: A Novel Approach to Computer Driven Tracing of Vessel Network 575
Annamaria Zaia, Pierluigi Maponi, Maria Marinelli, Anna Piantanelli, Roberto Giansanti, Roberto Murri

Automatic Extension of Korean Predicate-Based Sub-categorization Dictionary from Sense Tagged Corpora 585
Kyonam Choo, Seokhoon Kang, Hongki Min, Yoseop Woo

Information Fusion for Probabilistic Reasoning and Its Application to the Medical Decision Support Systems 593
Michal Wozniak

Robust Contrast Enhancement for Microcalcification in Mammography 602
Ho-Kyung Kang, Nguyen N. Thanh, Sung-Min Kim, Yong Man Ro

Track on Computational Methods

Exact and Approximate Algorithms for Two–Criteria Topological Design Problem of WAN with Budget and Delay Constraints 611
Mariusz Gola, Andrzej Kasprzak

Data Management with Load Balancing in Distributed Computing 621
Jong Sik Lee

High Performance Modeling with Quantized System 630
Jong Sik Lee

New Digit-Serial Systolic Arrays for Power-Sum and Division Operation in GF(2^m) 638
Won-Ho Lee, Keon-Jik Lee, Kee-Young Yoo

Generation of Unordered Binary Trees 648
Brice Effantin

A New Systolic Array for Least Significant Digit First Multiplication in $GF(2^m)$ 656
Chang Hoon Kim, Soonhak Kwon, Chun Pyo Hong, Hiecheol Kim

Asymptotic Error Estimate of Iterative Newton-Type Methods and Its Practical Application 667
Gennady Yu. Kulikov, Arkadi I. Merkulov

Numerical Solution of Linear High-Index DAEs 676
Mohammad Mahdi Hosseini

Fast Fourier Transform for Option Pricing: Improved Mathematical Modeling and Design of Efficient Parallel Algorithm 686
Sajib Barua, Ruppa K. Thulasiram, Parimala Thulasiraman

Global Concurrency Control Using Message Ordering of Group Communication in Multidatabase Systems 696
Aekyung Moon, Haengrae Cho

Applications of Fuzzy Data Mining Methods for Intrusion DetectionSystems 706
Jian Guan, Da-xin Liu, Tong Wang

Pseudo-Random Binary Sequences Synchronizer Based on Neural Networks 715
Jan Borgosz, Boguslaw Cyganek

Calculation of the Square Matrix Determinant: Computational Aspects and Alternative Algorithms 722
Antonio Annibali, Francesco Bellini

Differential Algebraic Method for Aberration Analysis of Electron Optical Systems 729
Min Cheng, Yilong Lu, Zhenhua Yao

Optimizing Symmetric FFTs with Prime Edge-Length 736
Edusmildo Orozco, Dorothy Bollman

A Spectral Technique to Solve the Chromatic Number Problem in Circulant Graphs 745
Monia Discepoli, Ivan Gerace, Riccardo Mariani, Andrea Remigi

A Method to Establish the Cooling Scheme in Simulated Annealing Like Algorithms 755
Héctor Sanvicente-Sánchez, Juan Frausto-Solís

Packing: Scheduling, Embedding, and Approximating Metrics 764
Hu Zhang

Track on Computational Science Education

Design Patterns in Scientific Software 776
Henry Gardner

Task Modeling in Computer Supported Collaborative Learning Environments to Adapt to Mobile Computing 786
Ana I. Molina, Miguel A. Redondo, Manuel Ortega

Computational Science and Engineering (CSE) Education: Faculty and Student Perspectives 795
Hasan Dağ, Gürkan Soykan, Şenol Pişkin, Osman Yaşar

Computational Math, Science, and Technology: A New Pedagogical Approach to Math and Science Education 807
Osman Yaşar

Track on Computer Modeling and Simulation

Resonant Tunneling Heterostructure Devices – Dependencies on Thickness and Number of Quantum Wells 817
Nenad Radulovic, Morten Willatzen, Roderick V.N. Melnik

Teletraffic Generation of Self-Similar Processes with Arbitrary Marginal Distributions for Simulation: Analysis of Hurst Parameters 827
Hae-Duck J. Jeong, Jong-Suk Ruth Lee, Hyoung-Woo Park

Design, Analysis, and Optimization of LCD Backlight Unit Using Ray Tracing Simulation 837
Joonsoo Choi, Kwang-Soo Hahn, Heekyung Seo, Seong-Cheol Kim

An Efficient Parameter Estimation Technique for a Solute Transport Equation in Porous Media 847
Jaemin Ahn, Chung-Ki Cho, Sungkwon Kang, YongHoon Kwon

HierGen: A Computer Tool for the Generation of Activity-on-the-Node Hierarchical Project Networks 857
Miguel Gutiérrez, Alfonso Durán, David Alegre, Francisco Sastrón

Macroscopic Treatment to Polymorphic E-mail Based Viruses 867
Cholmin Kim, Soung-uck Lee, Manpyo Hong

Making Discrete Games 877
Inmaculada García, Ramón Mollá

Speech Driven Facial Animation Using Chinese Mandarin Pronunciation Rules 886
Mingyu You, Jiajun Bu, Chun Chen, Mingli Song

Autonomic Protection System Using Adaptive Security Policy 896
Sihn-hye Park, Wonil Kim, Dong-kyoo Kim

A Novel Method to Support User's Consent in Usage Control for Stable Trust in E-business 906
Gunhee Lee, Wonil Kim, Dong-kyoo Kim

Track on Financial and Economical Modeling

No Trade under Rational Expectations in Economy (A Multi-modal Logic Approach) 915
Takashi Matsuhisa

A New Approach for Numerical Identification of Optimal Exercise Curve 926
Chung-Ki Cho, Sunbu Kang, Taekkeun Kim, YongHoon Kwon

Forecasting the Volatility of Stock Index Returns: A Stochastic Neural Network Approach 935
Chokri Slim

Track on Mobile Computing Systems

A New IP Paging Protocol for Hierarchical Mobile IPv6 945
Myung-Kyu Yi, Chong-Sun Hwang

Security Enhanced WTLS Handshake Protocol 955
Jin Kwak, Jongsu Han, Soohyun Oh, Dongho Won

An Adaptive Security Model for Heterogeneous Networks Using MAUT and Simple Heuristics 965
Jongwoo Chae, Ghita Kouadri Mostéfaoui, Mokdong Chung

A New Mechanism for SIP over Mobile IPv6 975
Pyung Soo Kim, Myung Eui Lee, Soohong Park, Young Kuen Kim

A Study for Performance Improvement of Smooth Handoff Using Mobility Management for Mobile IP 985
Kyu-Tae Oh, Jung-Sun Kim

A Fault-Tolerant Protocol for Mobile Agent 993
Guiyue Jin, Byoungchul Ahn, Ki Dong Lee

Performance Analysis of Multimedia Data Transmission with PDA over an Infrastructure Network 1002
Hye-Sun Hur, Youn-Sik Hong

A New Synchronization Protocol for Authentication in Wireless LAN Environment 1010
Hea Suk Jo, Hee Yong Youn

A Study on Secure and Efficient Sensor Network Management Scheme Using PTD 1020
Dae-Hee Seo, Im-Yeong Lee

Author Index 1029

New Techniques in Designing Finite Difference Domain Decomposition Algorithm for the Heat Equation

Shen Weidong and Yang Shulin

Institute of Applied Physics and Computational Mathematics
P.O. Box 8009-14, Beijing China
{yang_shulin,shen_weidong}@iapcm.ac.cn

Abstract. This paper presents a new technique in designing the finite difference domain decomposition algorithm for the heat-equation. The basic procedure is to define the finite difference schemes at the interface grid points with smaller time step $\Delta\bar{t} = \Delta t / m$ (m is a positive integer) by the classical explicit scheme. The stability region of the algorithm is expanded m times comparing with the classical explicit scheme, and the prior error estimates for the numerical solutions are obtained for some algorithms when $m = 2$ or $m = 3$. Numerical experiments on stability and accuracy are also presented.

1 Introduction

In recent decade and more, the parallel numerical methods for the heat equation have been studied. D.J. Evans [1], Zhang Bao-lin [2] have developed a class of alternating schemes in three time levels, which are the AGE(Alternating Group Explicit) and the ASE-I(Alternating Segment Explicit-Implicit) methods. Both of AGE and ASE-I methods are unconditionally stable and have the obvious property of parallelism, and the latter can be more accurate in practical computation. In the design of these two methods Saul'yev asymmetric schemes [5] have been used. C.N. Dawson [3] has developed the finite difference domain decomposition algorithm in two time levels, which can change the global implicit computation into the local ones by a novel technique of using the larger mesh spacing $H = Dh$ (D is a positive integer, h is the uniform mesh spacing) in explicit scheme at the interface points. The algorithm increases the stability bounds of classical explicit scheme by D^2 times, and its numerical solution also satisfies error estimate of $O(\Delta t + h^2)$ when the time step Δt satisfies $\Delta t \approx h^2 \approx H^3$. The technique has been further extended by using Saul'yev asymmetric schemes at a pair of interface points in recent work by Zhang Bao-lin [4] and then the algorithm has increased the stability bounds by $2D^2$ times and the similar error estimate $O(\Delta t + h^2)$ for the approximate solution has been obtained.

In this paper, we present a new technique by using smaller time step $\Delta\bar{t} = \Delta t / m$ (m is a positive integer) in the classical explicit scheme at the interface points. The algorithms designed with new techniques can increase the stability bounds of the classical explicit scheme by m times, and their numerical solution satisfies the similar error estimates to that in 3.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 1-10, 2004.
© Springer-Verlag Berlin Heidelberg 2004

The rest of this paper is organized as follows. In the next section, we construct some schemes respectively for $m = 2$ and $m = 3$ at the interface points. In Section 3, we define the domain decomposition algorithms with the schemes in Section 2, for which the convergence results of the numerical solutions are obtained. In section 4, some numerical examples are given to show the stability and the accuracy of the algorithms.

2 Schemes at Interface Points

Let $u(x,t)$ be the solution of the heat equation

$$\frac{\partial u}{\partial t} - \frac{\partial^2 u}{\partial x^2} = 0, \qquad x \in (0,1), \qquad t \in (0,T] \tag{1}$$

$$u(x,0) = u^0(x), \qquad x \in (0,1) \tag{2}$$

$$u(0,t) = u(1,t) = 0, \qquad t \in (0,T] \tag{3}$$

Firstly, the domain $(0,1)$ is decomposed into two sub-domains $(0,\bar{x})$ and $(\bar{x},1)$. Suppose N is a positive integer, $h = 1/N$, $x_i = ih$, $i = 0,1,\cdots,N$, and suppose here $\bar{x} = x_K > 0$ for some integer K. Let $\Delta t = T/M$, M is a positive integer, and suppose $t^n = n\Delta t$, $f_i^n = f(x_i, t^n)$ and define the difference operator

$$\partial_{t,\Delta t} f(t) = \frac{f(t) - f(t-\Delta t)}{\Delta t} \tag{4}$$

$$\partial^2_{x,h} f(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} \tag{5}$$

We will refer to points (x_i, t^n) as boundary points if $i = 0$ or N, or if $n = 0$. Similarly, we refer to them as interface points, if $x_i = \bar{x}$ and $n > 0$. Otherwise, they are interior points.

2.1 Explicit Scheme for the Case $m = 2$

Let $\Delta\bar{t} = \Delta t/2$, $t^{n+1/2} = t^n + \Delta\bar{t}$. As in Figure 1.

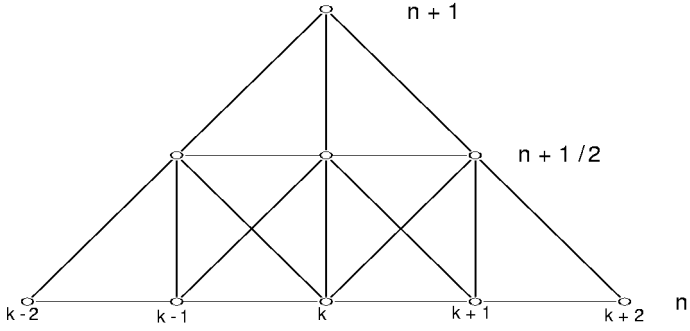


Fig. 1.

There are five grid points $k-2, k-1, k, k+1, k+2$ used to define explicit scheme respectively at additional time level $t^{n+1/2}$ and the t^{n+1}.
In detail, at first we use explicit schemes at points (x_k, t^{n+1})

$$U_i^{n+1} = \bar{r}U_{i+1}^{n+1/2} + (1-2\bar{r})U_i^{n+1/2} + \bar{r}U_{i-1}^{n+1/2} \tag{6}$$

where $\bar{r} = \Delta\bar{t}/h^2 = r/2$, $r = \Delta t/h^2$
Similarly, we can get the follow values,

$$U_{i+1}^{n+1/2} = \bar{r}U_{i+2}^{n} + (1-2\bar{r})U_{i+1}^{n} + \bar{r}U_i^{n} \tag{7}$$

$$U_i^{n+1/2} = \bar{r}U_{i+1}^{n} + (1-2\bar{r})U_i^{n} + \bar{r}U_{i-1}^{n} \tag{8}$$

$$U_{i+1}^{n+1/2} = \bar{r}U_i^{n} + (1-2\bar{r})U_{i-1}^{n} + \bar{r}U_{i-2}^{n} \tag{9}$$

Inserting (7)-(9) into (5), we can obtain the following scheme which will be used at the interface points

$$U_i^{n+1} = \bar{r}^2U_{i+2}^{n} + 2\bar{r}(1-2\bar{r})U_{i+1}^{n} + (2\bar{r}^2 + (1-2\bar{r})^2)U_i^{n}$$

$$+ 2\bar{r}(1-2\bar{r})U_{i-1}^{n} + \bar{r}^2U_{i-2}^{n} \tag{10}$$

Using Taylor expansion, we can easily get the truncation error

$$E_1 = \frac{\Delta t}{2}\frac{\partial^2 u}{\partial t^2}\bigg|_j^n - (\frac{\Delta t}{4} + \frac{h^2}{12})\frac{\partial^4 u}{\partial x^4}\bigg|_j^n + O(\Delta t^2 + h^3) \tag{11}$$

For convenience, we define operators L_1 for describing schemes (10) as follows

$$L_1 U_i^{n+1} = U_i^{n+1} - (\bar{r}^2 U_{i+2}^n + 2\bar{r}(1-2\bar{r})U_{i+1}^n$$

$$+ (2\bar{r}^2 + (1-2\bar{r})^2)U_i^n + 2\bar{r}(1-2\bar{r})U_{i-1}^n + \bar{r}^2 U_{i-2}^n) \tag{12}$$

2.2 Explicit Scheme for the Case $m = 3$

Let $\Delta\bar{t} = \Delta t/3$, $t^{n+1/3} = t^n + \Delta\bar{t}$, $t^{n+2/3} = t^{n+1/3} + \Delta\bar{t}$. As in Figure 2

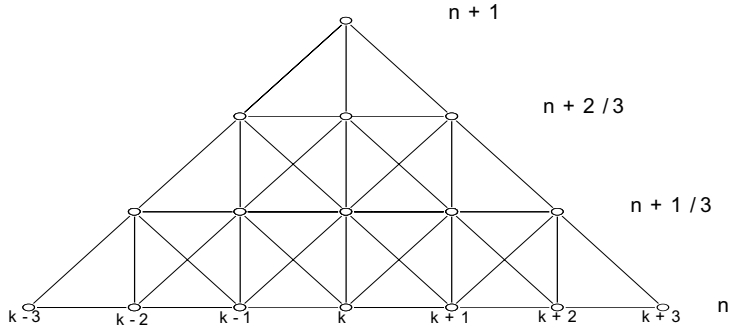


Fig. 2.

There are seven grid points $k-3, k-2, k-1, k, k+1, k+2, k+3$ used to define explicit scheme respectively at additional time level $t^{n+1/3}$, $t^{n+2/3}$ and the t^{n+1}. Similarly deduced as (10), have

$$U_i^{n+1} = \bar{r}^3 U_{i+3}^n + 3\bar{r}^2(1-2\bar{r})U_{i+2}^n + (3\bar{r}^3 + 3\bar{r}(1-2\bar{r})^2)U_{i+1}^n$$

$$+ ((1-2\bar{r})(6\bar{r}^2 + (1-2\bar{r})^2))U_i^n$$

$$+ (3\bar{r}^3 + 3\bar{r}(1-2\bar{r})^2)U_{i-1}^n + 3\bar{r}^2(1-2\bar{r})U_{i-2}^n + \bar{r}^3 U_{i-3}^n \tag{13}$$

where $\bar{r} = \Delta\bar{t}/h^2 = r/3$, $r = \Delta t/h^2$
Using Taylor expansion, we can easily get the truncation error

$$E = \frac{\Delta t}{2}\frac{\partial^2 u}{\partial t^2}\bigg|_j^n - (\frac{\Delta t}{3} + \frac{h^2}{12})\frac{\partial^4 u}{\partial x^4}\bigg|_j^n + O(\Delta t^2 + h^3) \tag{14}$$

For convenience, we define operator L_2 for describing schemes (10) as follows

$$L_2U_i^{n+1} = U_i^{n+1} - \bar{r}^3U_{i+3}^n + 3\bar{r}^2(1-2\bar{r})U_{i+2}^n + (3\bar{r}^3 + 3\bar{r}(1-2\bar{r})^2)U_{i+1}^n$$

$$+ ((1-2\bar{r})(6\bar{r}^2 + (1-2\bar{r})^2))U_i^n$$

$$+ (3\bar{r}^3 + 3\bar{r}(1-2\bar{r})^2)U_{i-1}^n + 3\bar{r}^2(1-2\bar{r})U_{i-2}^n + \bar{r}^3U_{i-3}^n \tag{15}$$

3 Designing and Analysis of Domain Decomposition Methods

After getting the schemes at interface points and the related operators L_1, L_2 which are defined in the previous section, we can design the following finite difference domain decomposition algorithms for the problem (1)-(3).

ALGORITHM I.

$U_i^n = u_i^n$ at boundary points (16)

$L_1U_i^n = 0$ at interface points $\bar{x}$ (17)

$LU_i^n = 0$ at interior points (18)

ALGORITHM II.

$U_i^n = u_i^n$ at boundary points (19)

$L_2U_i^n = 0$ at interface points $\bar{x}$ (20)

$LU_i^n = 0$ at interior points (21)

Here U_i^n is the numerical solution to u_i^n in (21). The purely implicit scheme is used and the operator L is defined as follows:

$$LU_i^n = \partial_{t,\Delta t}U_i^n - \partial_{x,h}^2U_i^n \tag{22}$$

One would expect that there be a constraint of the form

$$\Delta t \le mh^2/2 \tag{23}$$

Notice that in advancing the solution from time level $t = t^{n-1}$ to $t = t^n$ one first computes the value of U at the interface. This step requires a small amount of infor-

mation from each sub-domain. After the interface value has been computed, there are two completely separate backward difference problems to solve, which can be done in parallel. The prior error estimate of U in the Algorithm I is as follows:

Theorem 1. if $\Delta t \le h^2$, the numerical solution U_i^n of the Algorithms I, for solving the problem (1)-(3) satisfies

$$\max_{i,n} | u(x_i, t^n) - U_i^n | \le \frac{1}{8} C_0 [\Delta t + h^2 + 2h(\Delta t + h^2)] \tag{24}$$

$$C_0 = \max(\frac{1}{2} \| \partial^2 u / \partial t^2 \|, \quad \frac{1}{12} \| \partial^4 u / \partial x^4 \|) .$$

The proof of Theorem 1 relies on the following maximum principle

Lemma 1. Suppose that $\Delta t \le mh^2 / 2$ and that z_i^n satisfies the following relations:

$$z_i^n \le 0 \qquad \text{at boundary points} \tag{25}$$

$$L_j z_i^n \le 0 \qquad \text{at interface points } \bar{x} \tag{26}$$

$$L z_i^n \le 0 \qquad \text{at interior points} \tag{27}$$

where $j = 1,2$, Then for each i and n ,

$$z_i^n \le 0 \tag{28}$$

The proof of Theorem 1 relies on the following lemmas.

Lemma 2. Construct a discrete function

$$\beta_i = \begin{cases} \dfrac{1}{1-\bar{r}} h x_i (1-\bar{x}), & 0 \le x_i \le x_K = \bar{x} \\ \dfrac{1}{1-\bar{r}} h (1-x_i)\bar{x}, & x_K \le x_i \le 1 \end{cases} \tag{29}$$

Where $x_i = ih (i = 1,2,\cdots,N), \quad h = \dfrac{1}{N}; \quad \bar{r} = \Delta \bar{t} / h^2 = r/2 , r = \Delta t / h^2$,

then

$$L_1 \beta_K = \Delta t \tag{30}$$

when $i \ne K$,

$$L \beta_i = 0 \tag{31}$$

Lemma 3. Construct a discrete function

$$\beta_i = \begin{cases} \dfrac{1}{1-2\bar{r}+2\bar{r}^2} hx_i(1-\bar{x}), & 0 \le x_i \le x_K = \bar{x} \\ \dfrac{1}{1-2\bar{r}+2\bar{r}^2} h(1-x_i)\bar{x}, & x_K \le x_i \le 1 \end{cases} \tag{32}$$

Where $x_i = ih(i = 1,2,\cdots,N), \quad h = \dfrac{1}{N}; \quad \bar{r} = \Delta\bar{t}/h^2 = r/2\,, r = \Delta t/h^2$.

then

$$L_2\beta_K = \Delta t \tag{33}$$

when $i \neq K$,

$$L\beta_i = 0 \tag{34}$$

Proof of Theorem 1. Let $e_i^n = u_i^n - U_i^n$, then

$$e_i^n = 0 \qquad \text{at boundary points} \tag{35}$$

$$L_1 e_i^n = K_i^n \Delta t(\Delta t + h^2) \qquad \text{at interface points } \bar{x} \tag{36}$$

$$Le_i^n = K_i^n(\Delta t + h^2) \qquad \text{at interior points} \tag{37}$$

where

$$|K_i^n| \le C_0 \tag{38}$$

Construct a discrete function

$$\theta_i = \frac{1}{2} x_i(1-x_i)$$

then θ_i satisfies

$$\begin{aligned} &(i) \quad \theta_0 = \theta_N = 0 \\ &(ii) \quad when\ 0 < i < N, \quad L\theta_i = 1 \\ &(iii) \quad L_1\theta_k = \Delta t \\ &(iv) \quad 0 \le \theta_i \le \frac{1}{8} \end{aligned} \tag{39}$$

Furthermore, construct a discrete function β_i.

$$\beta_i = \begin{cases} \dfrac{1}{1-\bar{r}} hx_i(1-\bar{x}), & 0 \le x_i \le x_K = \bar{x} \\ \dfrac{1}{1-\bar{r}} h(1-x_i)\bar{x}, & x_K \le x_i \le 1 \end{cases} \tag{40}$$

β_i satisfies (30),(31) and

$$0 \le \beta_i < \frac{h}{2} \tag{41}$$

Suppose

$$z_i^n = e_i^n - \zeta_i \tag{42}$$

$$\zeta_i = C_2[\theta_i(\Delta t + h^2) + \beta_i(\Delta t + h^2)] \tag{43}$$

z_i satisfies the conditions in Lemma 1, so $z_i^n \le 0$,then $e_i^n \le \zeta_i$ .Similar we have $-e_i^n \le \zeta_i$ .therefore

$$|e_i^n| \le \zeta_i \le \frac{1}{8} C_2[\Delta t + h^2 + 2h(\Delta t + h^2)] \tag{44}$$

and Theorem 2 is proved.

Theorem 2. if $\Delta t \le 3h^2/2$ 时 , the numerical solution U_i^n of the Algorithms II, for solving the problem (1)-(3) satisfies

$$\max_{i,n} |u(x_i, t^n) - U_i^n| \le \frac{1}{8} C_0[\Delta t + h^2 + 2h(\Delta t + h^2)] \tag{45}$$

$C_0 = \max(\frac{1}{2} \| \partial^2 u / \partial t^2 \|, \quad \frac{1}{12} \| \partial^4 u / \partial x^4 \|)$.

The proof of Theorem 2 is similar to that for Theorem 1. By constructing a discrete function as following:

$$\beta_i = \begin{cases} \dfrac{1}{1-2\bar{r}+2\bar{r}^2} hx_i(1-\bar{x}), & 0 \le x_i \le x_K = \bar{x} \\ \dfrac{1}{1-2\bar{r}+2\bar{r}^2} h(1-x_i)\bar{x}, & x_K \le x_i \le 1 \end{cases}$$

4 Numerical Experiments

Take $u(x,0)=\sin\pi x$ in (2), then the exact solution of problem (1)-(3) is $u(x,t)=e^{-\pi^2 t}\sin\pi x$ Tables 1-2 list the results of the algorithm I and the implicit scheme, where $r=0.5, r=1.0$ and $r=1.5$. Tables 3-5 list the results of the algorithm II and the implicit scheme, where $r=1.0, r=1.5$ and $r=2.0$. In the tables, "implicit" is the classical implicit scheme. The new algorithms have the similar results in accuracy with the implicit scheme.

Table 1. r = 1.0, t = 0.5, dt = .63e^{3}, h = 0.25e^{-1}

x	Exact solution	numerical solution		relative error (100%)	
		implicit	algorithm.I	implicit	algorithm.I
0.1	.22224E-02	.22621E-02	.22572E-02	.17835E+01	.15649E+01
0.5	.71919E-02	.73202E-02	.73028E-02	.17835E+01	.15423E+01
0.9	.22224E-02	.22621E-02	.22572E-02	.17835E+01	.15649E+01

Table 2. r = 1.5, t = 0.5062, dt = .94e^{-3}, h = 0.25e^{-1}

x	Exact solution	numerical solution		relative error (100%)	
		implicit	algorithm.I	implicit	algorithm.I
0.1	.22087E-02	.22652E-02	.22506E-02	.25555E+01	.18954E+01
0.5	.71477E-02	.73303E-02	.72783E-02	.25555E+01	.18270E+01
0.9	.22087E-02	.22652E-02	.22506E-02	.25555E+01	.18954E+01

Table 3. r = 1.0, t = 0.5, dt = .63e^{-3}, h = 0.25e^{-1}

x	Exact solution	numerical solution		relative error (100%)	
		implicit	algorithm.II	implicit	algorithm.II
0.1	.22224E-02	.22621E-02	.22582E-02	.17835E+01	.16087E+01
0.5	.71919E-02	.73202E-02	.73063E-02	.17835E+01	.15983E+01
0.9	.22224E-02	.22621E-02	.22582E-02	.17835E+01	.16087E+01

Table 5. r = 1.5, t = 0.5062, dt = $.94e^{-3}$, h = $0.25e^{-1}$

x	Exact solution	numerical solution		relative error (100%)	
		implicit	algorithm.II	implicit	algorithm.II
0.1	.22087E-02	.22652E-02	.22587E-02	.25555E+01	.22635E+01
0.5	.71477E-02	.73303E-02	.73073E-02	.25555E+01	.22332E+01
0.9	.22087E-02	.22652E-02	.22587E-02	.25555E+01	.22635E+01

Table 6. r = 2.0, t = 0.5, dt = $.13e^{-2}$, h = $0.25e^{-1}$

x	Exact solution	numerical solution		relative error (100%)	
		implicit	algorithm.II	implicit	algorithm.II
0.1	.22224E-02	.22963E-02	.22885E-02	.33238E+01	.29736E+01
0.5	.71919E-02	.74309E-02	.74031E-02	.33238E+01	.29372E+01
0.9	.22224E-02	.22963E-02	.22885E-02	.33238E+01	.29736E+01

References

1. D.J. Evans, Alternating group explicit method for the diffusion equation, Appl. Math. Modelling, 19 (1985), 201-206
2. Zhang Bao-lin, An alternating segment explicit-implicit method for the diffusion equation, Chinese J. Num. Math Appl, 14:3 (1992), 27-37
3. C.N.Dawson, Qiang Du and T.F.Dupont, A finite difference domain decomposition algorithm for numerical solution of the heat equation, Math. Compt. 57 (1991), 63-71.
4. Zhang Baolin, Shen Weidong, Notes on Finite Difference Domain Decomposition Algorithm for the Solution of Heat Equation, Chinese J. Num. Math. & appl.24:3 (2002), 1-10
5. V.K. Saul'yev, Integration of equations of parabolic type by method of nets, New York, 1964.

A Fast Construction Algorithm for the Incidence Matrices of a Class of Symmetric Balanced Incomplete Block Designs

Ju-Hyun Lee[1], Sungkwon Kang[2⋆], and Hoo-Kyun Choi[3]

[1] Department of Mathematics & College of Pharmacy, Chosun University, Gwangju, 501-759, Korea
[2] Department of Mathematics, Chosun University, Gwangju, 501-759, Korea
sgkang@chosun.ac.kr
[3] College of Pharmacy, Chosun University, Gwangju, 501-759, Korea

Abstract. The theory of symmetric balanced incomplete block designs (BIBDs) has been applied in many research areas such as colored graphs, visual cryptography, distributed systems, communication networks, etc. In this paper, an explicit formula for a class of symmetric BIBDs is presented. Based on this formula, an efficient algorithm for constructing the incidence matrix of the design is developed. The incidence matrix contains all essential information of the design. The computational costs of the algorithm are $O(v)$ which are superior to those of $O(v^2)$ or $O(v\sqrt{v})$ by the conventional methods, where v is the number of objects or blocks.

1 Introduction

Let v, k, and λ be positive integers such that $v > k \geq 2$. A (v, k, λ)-*balanced incomplete block design*((v, k, λ)-BIBD) is a pair $(X, \mathcal{A})$ such that the following conditions are satisfied[1,9].

(i) X is a set of v elements called *objects*.
(ii) $\mathcal{A}$ is a collection of subsets of X called *blocks*.
(iii) Each block contains k objects.
(iv) Every pair of distinct objects is contained in exactly λ blocks.

The condition (iv) is the "balance" property. A BIBD is called an "incomplete" block design due to the condition that $k < v$. Also, note that a BIBD may contain repeated blocks if $\lambda > 1$, which is why we refer to a collection of blocks rather than a set. In a (v, k, λ)-BIBD, every object occurs in exactly $r = \frac{\lambda(v-1)}{k-1}$ blocks, and the design has exactly $b = \frac{vr}{k}$ blocks. Sometimes we use the notation (v, b, r, k, λ)-BIBD if we want to record the values of all the five parameters. A (v, b, r, k, λ)-BIBD can be described by the *incidence matrix* $\mathcal{M}$. It is a $v \times b$ zero-one matrix, i.e., its entries are 0 and 1. The rows and columns of the matrix correspond to the objects and the blocks, respectively. The entry in the $i-th$ row and the $j-th$ column of $\mathcal{M}$ is 1 if the block B_j contains the object x_i

⋆ The corresponding author

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 11–19, 2004.
© Springer-Verlag Berlin Heidelberg 2004

and is 0 otherwise. Thus, the incidence matrix contains all essential information of the design. If the number of objects is the same as that of blocks, i.e., $v = b$, the (v, k, λ)-BIBD is called *symmetric*. The symmetric (v, k, λ)-BIBDs have been applied in many areas such as colored graphs, visual cryptography schemes, distributed systems, communication networks, etc.[1,3,4,7,8]. In any symmetric (v, k, λ)-BIBD, $k = r$, i.e., the number of the objects in each block is the same as that of the blocks containing a given object. If $\mathcal{M}$ is the incidence matrix of a symmetric (v, k, λ)-BIBD, the matrix obtained by any exchanges in the rows or the columns of $\mathcal{M}$ produces another symmetric (v, k, λ)-BIBD which is isomorphic to the original BIBD. Hence, the number of the symmetric (v, k, λ)-BIBDs generated by exchanging the rows or the columns of $\mathcal{M}$ is order of $O((v!)^2)$. Once a design is obtained, for example, a secured communication protocol in cryptology can be developed by rearranging the order of the objects and the blocks due to this extreme complexity.

In this paper, we consider the class of symmetric BIBDs with $v = q^2 + q + 1, k = q+1$, and $\lambda = 1$. It is known that the symmetric $(q^2+q+1, q+1, 1)$-BIBDs exist for any prime q. The members of this class of (v, k, λ)-BIBDs are called *finite projective planes of order q*. The incidence matrices representing the symmetric $(q^2 + q + 1, q + 1, 1)$-BIBDs have been used for constructing the congestion free networks[2], the two out of $q^2 + q + 1$ schemes in visual cryptography[3,5,7], a conference key distribution system[1], and a message load balancing scheme in a distributed system[6]. An issue in these designs is to develop fast construction algorithms. In this paper, by systematic approaches and careful investigation of the relations between objects and blocks, we derive an explicit formula for the design and a fast construction algorithm with time complexity $O(v) = O(q^2)$ which are superior to those of $O(v^2)$ or $O(v\sqrt{v})$ by the conventional methods.

Throughout this paper, $\mathbb{Z}_q = \{0, 1, 2, \cdots, q-1\}$ is the finite field obtained by taking the modulus q to the set of all nonnegative integers, $< x >$ denotes the generator by a vector x, and, for a given set or a collection of elements X, $|X|$ denotes the number of elements.

2 Explicit Design Formula

In this section, we consider a systematic construction method for a class of the symmetric $(q^2 + q + 1, q + 1, 1)$-BIBDs, where q is a prime number. Consider the following three-dimensional vector space

$$V = (\mathbb{Z}_q)^3 = \{(a, b, c) | a, b, c \in \mathbb{Z}_q\} \tag{1}$$

over the scalar field $\mathbb{Z}_q$, and for the one-dimensional subspaces of V, let

$$\begin{aligned} x_1 &=< (1,0,0) >= \{a(1,0,0) | a \in \mathbb{Z}_q\}, \\ x_{i+2} &=< (i,0,1) >= \{a(i,0,1) (\mathrm{mod}\ q) | a \in \mathbb{Z}_q\} \text{ for } 0 \le i \le q-1, \\ x_{q(i+1)+(j+2)} &=< (j,1,i) > \\ &= \{a(j,1,i) (\mathrm{mod}\ q) | a \in \mathbb{Z}_q\} \text{ for } 0 \le i, j \le q-1. \end{aligned} \tag{2}$$

For the two-dimensional subspaces of V, let

$$\begin{aligned}
\widetilde{B}_1 &= <(1,0,0)> + <(0,0,1)> \\
&= \{(a,0,b) | a,b \in \mathbb{Z}_q\}, \\
\widetilde{B}_{i+2} &= <(1,0,0)> + <(0,1,i)> \\
&= \{(a,b,bi) (\mathrm{mod}\, q) | a,b \in \mathbb{Z}_q\},\ 0 \le i \le q-1, \\
\widetilde{B}_{q(i+1)+(j+2)} &= <(i,0,1)> + <(j,1,0)> \\
&= \{(ai+bj,b,a) (\mathrm{mod}\, q) | a,b \in \mathbb{Z}_q\},\ 0 \le i,j \le q-1.
\end{aligned} \tag{3}$$

Let

$$X = \{x_1, x_2, \cdots, x_{q^2+q+1}\}, \tag{4}$$

$$\widetilde{\widetilde{\mathcal{B}}} = \{\widetilde{B}_1, \widetilde{B}_2, \cdots, \widetilde{B}_{q^2+q+1}\}, \tag{5}$$

where $\{x_i\}$ and $\{\widetilde{B}_j\}$ are defined as in (2)-(3). Then X and $\widetilde{\widetilde{\mathcal{B}}}$ are the collections of the one-dimensional subspaces $\{x_i\}$ and the two-dimensional subspaces $\{\widetilde{B}_j\}$ of V, respectively. Then we have the following.

Theorem 1. *Let $X = \{x_i\}$ and $\widetilde{\widetilde{\mathcal{B}}} = \{\widetilde{B}_j\}$ be given by (4)-(5). Then we have the following properties.*
(i) $|X| = q^2 + q + 1$.
(ii) $|\widetilde{\widetilde{\mathcal{B}}}| = q^2 + q + 1$.
(iii) Every pair of distinct x and y in X is contained in only one $\widetilde{B} \in \widetilde{\widetilde{\mathcal{B}}}$.
(iv) Every $\widetilde{B} \in \widetilde{\widetilde{\mathcal{B}}}$ contains exactly $q+1$ elements of X.

Proof. (i) It is easy to see that $x_i \cap x_j = \{(0,0,0)\}$ for any $i \neq j$. Hence, by the definition of X in (4), $|X| = q^2 + q + 1$. (ii) It suffices to show that $\widetilde{B}_i \neq \widetilde{B}_j$ for any $i \neq j$, where $\widetilde{B}_i, \widetilde{B}_j \in \widetilde{\widetilde{\mathcal{B}}}$. By the definition of $\{\widetilde{B}_i\}$ in (3), it is clear that $\widetilde{B}_1 \neq \widetilde{B}_{i+2}$ for all $0 \le i \le q-1$ and $\widetilde{B}_1 \neq \widetilde{B}_{q(i+1)+(j+2)}$ for all $0 \le i,j \le q-1$. Next, we prove that $\widetilde{B}_{i+2} \neq \widetilde{B}_{q(s+1)+(t+2)}$ for all $0 \le i,s,t \le q-1$. If $i \neq 0$, any element of $\widetilde{B}_{i+2}$ has the form $(a,b,bi(\mathrm{mod}\, q))$ for some a and b in $\mathbb{Z}_q$. Note that, for each t, $0 \le t \le q-1$, $(t,1,0) \in \widetilde{B}_{q(s+1)+(t+2)}$ for all $0 \le s \le q-1$. If $(t,1,0) = (a,b,bi(\mathrm{mod}\, q))$ for some $a,b \in \mathbb{Z}_q$, we have $t = a$, $b = 1$, and $bi(\mathrm{mod}\, q) = 0$. Thus, i must be 0. Since we assumed that $i \neq 0$, it is impossible. Therefore, for every $i \neq 0$, $\widetilde{B}_{i+2} \neq \widetilde{B}_{q(s+1)+(t+2)}$ for all $0 \le s,t \le q-1$. Let $i = 0$. Then for each s, $0 \le s \le q-1$, $(s,0,1) \in \widetilde{B}_{q(s+1)+(t+2)}$ for all $0 \le t \le q-1$. But, $(s,0,1) \notin \widetilde{B}_2 = \{(a,b,0) | a,b \in \mathbb{Z}_q\}$. Hence, for any i, $0 \le i \le q-1$, $\widetilde{B}_{i+2} \neq \widetilde{B}_{q(s+1)+(t+2)}$ for all $0 \le s,t \le q-1$. Now, suppose that $\widetilde{B}_{i+2} = \widetilde{B}_{j+2}$ for some $i \neq j$. Then $(a,b,bi(\mathrm{mod} q)) = (c,d,dj(\mathrm{mod} q))$ for some $a,b,c,d \in \mathbb{Z}_q$. Thus, $a = c$, $b = d$, and $bi(\mathrm{mod}\, q) = dj(\mathrm{mod}\, q)$. Hence, $bi(\mathrm{mod}\, q) = bj(\mathrm{mod}\, q)$, and i must be equal to j which contradicts the assumption. Therefore, $\widetilde{B}_{i+2} \neq \widetilde{B}_{j+2}$ for any $i \neq j$. Finally, we consider any two-dimensional subspaces $\widetilde{B}_{q(i+1)+(j+2)}$ and $\widetilde{B}_{q(s+1)+(t+2)}$. For

any fixed i and j, $0 \le i, j \le q-1$, let $z = ((ai+bj)(\mathrm{mod}\, q), b, a)$ be an element of $\widetilde{B}_{q(i+1)+(j+2)}$. If $z \in \widetilde{B}_{q(s+1)+(t+2)}$ for some $s, t \in \mathbb{Z}_q$, $z = ((cs+dt)(\mathrm{mod}\, q), d, c)$ for some $c, d \in \mathbb{Z}_q$, i.e., $((ai+bj)(\mathrm{mod}\, q), b, a) = ((cs+dt)(\mathrm{mod}\, q), d, c)$. Thus, we have $(ai+bj)(\mathrm{mod}\, q) = (cs+dt)(\mathrm{mod}\, q)$, $b = d$, and $a = c$. Hence, $(ci+dj)(\mathrm{mod}\, q) = (cs+dt)(\mathrm{mod}\, q)$. If $i = s$, we have $dj(\mathrm{mod}\, q) = dt(\mathrm{mod}\, q)$, and, thus, $j = t$. In this case, we must have $(i, j) = (s, t)$. Assume that $i \neq s$. If $j = t$, for any s, $0 \le s \le q-1$, $(s, 0, 1) \in \widetilde{B}_{q(s+1)+(t+2)}$ for all $0 \le t \le q-1$. But, $(s, 0, 1) \notin \widetilde{B}_{q(i+1)+(j+2)}$ for all i and j, $0 \le i, j \le q-1$, with $i \neq s$, since if $(s, 0, 1) = ((ai+bj)(\mathrm{mod} q), b, a)$ for some $a, b \in \mathbb{Z}_q$, then $(ai+bj)(\mathrm{mod} q) = s$, $b = 0$, and $a = 1$ so that we must have $i = s$ which contradicts the assumption $i \neq s$. Suppose that $j \neq t$. Then for any s, $0 \le s \le q-1$, $(t, 1, 0) \in \widetilde{B}_{q(s+1)+(t+2)}$ for all $0 \le t \le q-1$. If $(t, 1, 0) \in \widetilde{B}_{q(i+1)+(j+2)}$ for some i and j, $0 \le i, j \le q-1$, then $(t, 1, 0) = ((ai+bj)(\mathrm{mod}\, q), b, a)$ for some $a, b \in \mathbb{Z}_q$. Hence, $(ai+bj)(\mathrm{mod}\, q) = t$, $b = 1$, and $a = 0$, i.e., $j = t$ which contradicts the assumption $j \neq t$. Thus, $(t, 1, 0) \notin \widetilde{B}_{q(i+1)+(j+2)}$ for all i, j with $j \neq t$. Hence, for any i, j, s, and t with $(i, j) \neq (s, t)$, $\widetilde{B}_{q(i+1)+(j+2)} \neq \widetilde{B}_{q(s+1)+(t+2)}$. Therefore, for any $\widetilde{B}_i$ and $\widetilde{B}_j$ in $\widetilde{\widetilde{\mathcal{B}}}$ with $i \neq j$, $\widetilde{B}_i \neq \widetilde{B}_j$. By (5), $|\widetilde{\widetilde{\mathcal{B}}}| = q^2+q+1$. (iii) Note that any pair of distinct one-dimensional subspaces x and y in X generates a unique two-dimensional subspace of V. Since any $\widetilde{B}_i$ and $\widetilde{B}_j$, $i \neq j$, in $\widetilde{\widetilde{\mathcal{B}}}$ are all different, any pair of distinct one-dimensional subspaces is contained in only one $\widetilde{B} \in \widetilde{\widetilde{\mathcal{B}}}$. (iv) Let $\widetilde{B}$ be any element of $\widetilde{\widetilde{\mathcal{B}}}$. Then $|\widetilde{B} \setminus \{(0,0,0)\}| = q^2-1$. Since $\{x \setminus \{(0,0,0)\} | x \subseteq \widetilde{B}\}$ forms a partition of $\widetilde{B} \setminus \{(0,0,0)\}$ and for any $x \in X$, $|x \setminus \{(0,0,0)\}| = q-1$, the number of equivalence classes in $\widetilde{B} \setminus \{(0,0,0)\}$ is $\frac{q^2-1}{q-1} = q+1$. Thus, $|\{x \in X | x \subseteq \widetilde{B}\}| = q+1$. This completes the proof. □

Note that the number of all the one-dimensional subspaces of $V = (\mathbb{Z}_q)^3$ and the number of all the two-dimensional subspaces of V are q^2+q+1. By Theorem 1, $\{x_i\}$ and $\{\widetilde{B}_j\}$ in (2)-(3) classify all the one-dimensional and the two-dimensional subspaces of V. Thus, the collections X and $\widetilde{\widetilde{\mathcal{B}}}$ in (4)-(5) become sets. Therefore, we have the following theorem.

Theorem 2. *Let X and $\widetilde{\widetilde{\mathcal{B}}}$ be given as in (4)-(5). Then X and $\widetilde{\widetilde{\mathcal{B}}}$ are the set of all the one-dimensional and the two-dimensional subspaces of V, respectively.*

The following two theorems show the inclusion relations between the one-dimensional subspaces $\{x_i\}$ and the two-dimensional subspaces $\{\widetilde{B}_j\}$ of V.

Theorem 3. *Let $\{x_i\}$ and $\{\widetilde{B}_j\}$ be defined as in (2) and (3), respectively. Then we have the following relations.*
(i) $x_1 \subseteq \widetilde{B}_1$, $\widetilde{B}_{m+2}$, $0 \le m \le q-1$.
(ii) For each i, $0 \le i \le q-1$, $x_{i+2} \subseteq \widetilde{B}_1$, $\widetilde{B}_{q(i+1)+(m+2)}$, $0 \le m \le q-1$.
(iii) For each i *and* j, $0 \le i, j \le q-1$,
$x_{q(i+1)+(j+2)} \subseteq \widetilde{B}_{i+2}$, $\widetilde{B}_{q(l+1)+((j-il)(mod\, q)+2)}$, $0 \le l \le q-1$.

Proof. (i) Since $x_1 = \{a(1,0,0)|a \in \mathbb{Z}_q\}$, $\widetilde{B}_1 = \{a(1,0,0) + b(0,0,1)|a,b \in \mathbb{Z}_q\}$, and $\widetilde{B}_{m+2} = \{(a(1,0,0) + b(0,1,m))(\mathrm{mod}\ q)|a,b \in \mathbb{Z}_q\}$, it is clear that $x_1 \subseteq \widetilde{B}_1, \widetilde{B}_{m+2}$ for $0 \le m \le q-1$. (ii) Let i, $0 \le i \le q-1$, be fixed. Since $x_{i+2} = \{a(i,0,1)(\mathrm{mod}\ q)|a \in \mathbb{Z}_q\}$ and $\widetilde{B}_1 = \{a(1,0,0) + b(0,0,1)|a,b \in \mathbb{Z}_q\}$, $x_{i+2} = \{(ai,0,a)(\mathrm{mod}\ q)|a \in \mathbb{Z}_q\}$ and, hence, $\widetilde{B}_1 = \{(a,0,b)|a,b \in \mathbb{Z}_q\} \supseteq x_{i+2}$. On the other hand, since $\widetilde{B}_{q(i+1)+(m+2)} = \{(a(i,0,1) + b(m,1,0))(\mathrm{mod}\ q)|a,b \in \mathbb{Z}_q\}$, it is clear that $x_{i+2} \subseteq \widetilde{B}_{q(i+1)+(m+2)}$ for $0 \le m \le q-1$. (iii) Let i and j, $0 \le i,j \le q-1$, be fixed. Since $x_{q(i+1)+(j+2)} = \{a(j,1,i)(\mathrm{mod}\ q)|a \in \mathbb{Z}_q\}$ and $\widetilde{B}_{i+2} = \{(l(1,0,0) + b(0,1,i))(\mathrm{mod}\ q)|l,b \in \mathbb{Z}_q\} = \{(l,b,bi)(\mathrm{mod}\ q)|l,b \in \mathbb{Z}_q\}$, if $l = bj(\mathrm{mod}\ q)$, $\{(l,b,bi)(\mathrm{mod}\ q)|l,b \in \mathbb{Z}_q\} = \{b(j,1,i)(\mathrm{mod}\ q)|b \in \mathbb{Z}_q\} = x_{q(i+1)+(j+2)}$. Hence, $x_{q(i+1)+(j+2)} \subseteq \widetilde{B}_{i+2}$. To prove that $x_{q(i+1)+(j+2)} \subseteq \widetilde{B}_{q(l+1)+((j-il)(\mathrm{mod}\ q)+2)}$ for $0 \le l \le q-1$, let i and j, $0 \le i,j \le q-1$, be fixed. Recall that $x_{q(i+1)+(j+2)} = \{c(j,1,i)(\mathrm{mod}\, q)|c \in \mathbb{Z}_q\}$ and $\widetilde{B}_{q(l+1)+(m+2)} = \{(a(l,0,1) + b(m,1,0))(\mathrm{mod} q)|a,b \in \mathbb{Z}_q\}$, $0 \le l,m \le q-1$. Hence, for each l and m, $0 \le l,m \le q-1$, $\widetilde{B}_{q(l+1)+(m+2)} = \{(al+bm,b,a)(\mathrm{mod}\ q)|a,b \in \mathbb{Z}_q\}$. If $(al+bm)(\mathrm{mod}\ q) = bj(\mathrm{mod}\ q)$ and $a = bi(\mathrm{mod}\ q)$, $(al+bm,b,a)(\mathrm{mod}\ q) = (bj,b,bi)(\mathrm{mod}\ q) = b(j,1,i)(\mathrm{mod}\ q)$. On the other hand, if $(al+bm)(\mathrm{mod}\ q) = bj(\mathrm{mod}\ q)$ and $a = bi(\mathrm{mod}\ q)$, $(al+bm)(\mathrm{mod}\ q) = (bil+bm)(\mathrm{mod}\ q) = b(il+m)(\mathrm{mod}\ q) = bj(\mathrm{mod}\ q)$.

Thus, for each l, $0 \le l \le q-1$, if we choose m such that $(il+m)(\mathrm{mod}\, q) = j$, i.e., $m = (j-il)(\mathrm{mod}\ q)$, then $\widetilde{B}_{q(l+1)+(m+2)} \supseteq x_{q(i+1)+(j+2)}$. This completes the proof. □

Theorem 4. *Let $\{x_i\}$ and $\{\widetilde{B}_j\}$ be given as in (2) and (3). Then we have the following.*
(i) $\widetilde{B}_1 \supseteq x_1, x_{m+2}$, $0 \le m \le q-1$.
(ii) For each i, $0 \le i \le q-1$,
$\widetilde{B}_{i+2} \supseteq x_1$, $x_{q(i+1)+(m+2)}$, $0 \le m \le q-1$.
(iii) For each i, j, $0 \le i,j \le q-1$,
$\widetilde{B}_{q(i+1)+(j+2)} \supseteq x_{i+2}$, $x_{q(l+1)+((j+il)(mod\, q)+2)}$, $0 \le l \le q-1$.

Proof. From Theorem 3, (i) and (ii) are clear.
(iii) Since $\widetilde{B}_{q(i+1)+(j+2)} = \{(a(i,0,1)+b(j,1,0))(\mathrm{mod} q)|a,b \in \mathbb{Z}_q\}$, it is clear that $\widetilde{B}_{q(i+1)+(j+2)} \supseteq x_{i+2}$. To prove that $B_{q(i+1)+(j+2)} \supseteq x_{q(l+1)+((j+il)(\mathrm{mod}\, q)+2)}$ for $0 \le l \le q-1$, let i and j, $0 \le i,j \le q-1$, be fixed. Recall that $\widetilde{B}_{q(i+1)+(j+2)} = \{(ai+bj,b,a)(\mathrm{mod}\, q)|a,b \in \mathbb{Z}_q\}$ and $x_{q(l+1)+(m+2)} = \{c(m,1,l)(\mathrm{mod}\, q)|c \in \mathbb{Z}_q\}$, $0 \le l,m \le q-1$. Hence, for each l and m, $0 \le l,m \le q-1$, if $(ai+bj)(\mathrm{mod}\, q) = bm(\mathrm{mod}\ q)$ and $a = bl(\mathrm{mod}\ q)$, $(ai+bj,b,a)(\mathrm{mod}\ q) = (bm,b,bl)(\mathrm{mod}\ q) = b(m,1,l)(\mathrm{mod}\ q)$. On the other hand, if $(ai+bj)(\mathrm{mod}\ q) = bm(\mathrm{mod}\ q)$ and $a = bl(\mathrm{mod}\ q)$, $(ai+bj)(\mathrm{mod}\ q) = b(il+j)(\mathrm{mod}\ q) = bm(\mathrm{mod}\ q)$. Thus, for each l, $0 \le l \le q-1$, if we choose m such that $m = (j+il)(\mathrm{mod}\ q)$, then $\widetilde{B}_{q(i+1)+(j+2)} \supseteq x_{q(l+1)+(m+2)}$. This completes the proof. □

We now define the set of blocks $\mathcal{B}$ by

$$\mathcal{B} = \{B_1, B_2, \cdots, B_{q^2+q+1}\}, \tag{6}$$

where

$$\begin{aligned} &B_1 = \{x_1, x_{m+2} \mid 0 \le m \le q-1\}, \\ &B_{i+2} = \{x_1, x_{q(i+1)+(m+2)} \mid 0 \le m \le q-1\},\ 0 \le i \le q-1, \\ &B_{q(i+1)+(j+2)} = \{x_{i+2}, x_{q(l+1)+((j+il)(\mathrm{mod}\ q)+2)} \mid 0 \le l \le q-1\}, \\ &\qquad 0 \le i, j \le q-1. \end{aligned} \tag{7}$$

Then, by Theorems 1-4, we have the following theorem.

Theorem 5. *Let X and $\mathcal{B}$ be given as in (4) and (6)-(7), respectively. Then $(X, \mathcal{B})$ becomes a symmetric $(q^2+q+1, q+1, 1)$-BIBD.*

Remark 1. (i) It is not known whether or not all projective planes of prime order have a vector space representation(this is a long standing open problem in finite geometry). (ii) In (7), for each given pair (i, j), $0 \le i, j \le q-1$, the calculations $q(l+1)+((j+il)(\mathrm{mod}\ q)+2)$, $0 \le l \le q-1$, are required. These computations are main obstacles in the design $(X, \mathcal{B})$ even though we have the explicit formula (7). Due to these obstacles, the total computational costs for the incidence matrix of the design are $3q^3$ in the order of q^3. Therefore, we need an efficient algorithm for handling those obstacles.

3 Derivation of Algorithm

In this section, we derive an efficient algorithm with the computational costs $O(q^2)$ for the incidence matrix of the design $(X, \mathcal{B})$ described in Section 2. To develop the algorithm, we will define the "position" matrix and the "cyclic extension" matrix. The matrices are obtained from the modulo q multiplication table.

For a given prime number q, let IJ be the $q \times q$ matrix defined by

$$\mathrm{IJ}(i, j) = ij(\mathrm{mod}\, q) \tag{8}$$

for $0 \le i, j \le q-1$, where $ij(\mathrm{mod}\, q)$ is the remainder of the multiplication ij after division by q. Note that the indices of the matrix IJ start from 0 instead of 1. Then we have the following properties.

Lemma 1. *(i) For all $0 \le i, j \le q-1$, $IJ(j, i) = IJ(i, j)$.*
(ii) For $1 \le i, j \le q-1$, $IJ(i, q-j) = q - IJ(i, j)$ and $IJ(q-i, j) = q - IJ(i, j)$.
(iii) For $1 \le i, j \le q-1$, $IJ(q-i, q-j) = IJ(i, j)$.

Remark 2. The computational costs for constructing the matrix IJ are $2q^2$(q^2 multiplications and q^2 divisions).

Define the "position" matrix PA by

$$\mathrm{PA}(i, \mathrm{IJ}(i,j)) = j,\ 1 \le i \le q-1,\ 0 \le j \le q-1, \tag{9}$$

and the "cyclic extension" matrix σ by

$$\sigma(i,j) = \begin{cases} \mathrm{IJ}(i,j), & 1 \le i \le q-1,\ 0 \le j \le q-1, \\ \mathrm{IJ}(i,j-q), & 1 \le i \le q-1,\ q \le j \le 2q-2. \end{cases} \tag{10}$$

Then PA and σ become a $(q-1) \times q$ matrix and a $(q-1) \times (2q-1)$ matrix, respectively. As we see in the definitions (9) and (10), we do not need any extra multiplications, divisions, or logical operations such as "if" statements to obtain the matrices PA and σ except those for constructing IJ. The position matrix PA and the extension matrix σ have the following relation.

Theorem 6. *For each given pair* (i,j), $1 \le i \le q-1$, $0 \le j \le q-1$,

$$(j+il)(mod\, q) = \sigma(i, PA(i,j)+l),\ 0 \le l \le q-1, \tag{11}$$

where PA and σ *are given by (9) and (10), respectively.*

Proof. Let each pair (i,j), $1 \le i \le q-1$, $0 \le j \le q-1$, be given. Since $\mathrm{PA}(i,j)$ indicates the column index of the i-th row of IJ containing the given number j, $\mathrm{IJ}(i, \mathrm{PA}(i,j)) = j$. Let $\mathrm{PA}(i,j) = t$. Then $\mathrm{IJ}(i,t) = j = it (\mathrm{mod}\, q)$. Thus, for $0 \le l \le q-1$, $(j+il)(\mathrm{mod}\, q) = (it(\mathrm{mod}\, q) + il)(\mathrm{mod}\, q) = i(t+l)(\mathrm{mod}\, q) = \mathrm{IJ}(i, (t+l)(\mathrm{mod}\, q))$.

Note that $0 \le t+l \le 2q-2$. If $0 \le t+l \le q-1$, $\mathrm{IJ}(i,(t+l)(\mathrm{mod}\, q)) = \mathrm{IJ}(i, t+l) = \sigma(i, t+l)$, and if $q \le t+l \le 2q-2$, $\mathrm{IJ}(i,(t+l)(\mathrm{mod}\, q)) = \mathrm{IJ}(i,(t+l)-q) = \sigma(i,t+l)$ by (10). Therefore, $(j+il)(\mathrm{mod}\, q) = \sigma(i, \mathrm{PA}(i,j)+l)$. □

Remark 3. (i) In (7), the multiplications $q(l+1)$, $0 \le l \le q-1$, can be performed by addition iterations. (ii) For $i = 0$ and $0 \le j \le q-1$, we can find the rows and columns of the incidence matrix $\mathcal{M}$ directly. On the other hand, by Theorem 6, for each i and j, $1 \le i \le q-1$, $0 \le j \le q-1$, $(j+il)(\mathrm{mod}\, q)$, $0 \le l \le q-1$, are obtained by σ and PA. Thus, we do not need any extra multiplications or divisions to calculate $(j+il)(\mathrm{mod}\, q)$.

We are ready to state the $O(v) = O(q^2)$ algorithm for constructing the incidence matrix $\mathcal{M}$ of the design $(X, \mathcal{B})$ in (7).

Algorithm for constructing $\mathcal{M}$ of $(X, \mathcal{B})$

Step 0. Set $\mathcal{M}$ be a $(q^2+q+1) \times (q^2+q+1)$ zero matrix.
Step 1. Let $\mathcal{M}(s,1) = 1$ for $1 \le s \le q+1$.
Step 2. Let $I = 0$.
Step 3. For $0 \le i \le q-1$, do Step 3.1 - Step 3.4.
  Step 3.1. $I = I + q$.
  Step 3.2. $t = i + 2$.

Step 3.3. Do $\mathcal{M}(1,t)=1$.
Step 3.4. Do $\mathcal{M}(I+(m+2),t)=1$ for $0\le m\le q-1$.
Step 4. For $0\le j\le q-1$, do Step 4.1 - Step 4.4.
Step 4.1. $t=j+2$.
Step 4.2. Do $\mathcal{M}(2,q+t)=1$.
Step 4.3. Let $L=0$.
Step 4.4. For $0\le l\le q-1$, do Step 4.4.1 - Step 4.4.2.
Step 4.4.1. $L=L+q$.
Step 4.4.2. Do $\mathcal{M}(L+t,q+t)=1$.
Step 5. Let $I=q$.
Step 6. For $1\le i\le q-1$, do Step 6.1 - Step 6.5.
Step 6.1. Construct the i-th row M of IJ by (8).
Step 6.2. Construct the i-th row P of PA from M by (9).
Step 6.3. Construct the i-th row T of σ from M by (10).
Step 6.4. $I=I+q$.
Step 6.5. For $0\le j\le q-1$, do Step 6.5.1 - Step 6.5.5.
Step 6.5.1. $t=j+2$.
Step 6.5.2. Do $\mathcal{M}(i+2,I+t)=1$.
Step 6.5.3. $p=P(j)$.
Step 6.5.4. Let $L=0$.
Step 6.5.5. For $0\le l\le q-1$, do Step 6.5.5.1 - Step 6.5.5.3.
Step 6.5.5.1. $L=L+q$.
Step 6.5.5.2. $p=p+l$.
Step 6.5.5.3. Do $\mathcal{M}(L+(T(p)+2),I+t)=1$.

Remark 4. In the above algorithm, we do not form the matrix IJ. Instead, each row vector M of IJ is constructed during the iteration process. The position vector P and the cyclic extension vector T are obtained directly from this M without any multiplications or divisions. Therefore, the total time complexity of the algorithm is $O(v)=O(q^2)$.

Example 1. In this example, we consider the implementation of the algorithm with $q=3$ for simplicity. Note that the number of objects or blocks is $q^2+q+1=13$, and that the number of the objects contained in each block is the same as that of the blocks containing a given object, i.e., $q+1=4$. Thus, the incidence matrix $\mathcal{M}$ of the symmetric (13,4,1)-BIBD is a 13×13 matrix. From each step of the algorithm, the components of $\mathcal{M}$ are obtained. Finally, the incidence matrix $\mathcal{M}$ becomes:

$$
\begin{array}{c}
 & \begin{array}{ccccccccccccc} B_1 & B_2 & B_3 & B_4 & B_5 & B_6 & B_7 & B_8 & B_9 & B_{10} & B_{11} & B_{12} & B_{13} \end{array} \\
\begin{array}{c} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \\ x_{13} \end{array} &
\left(\begin{array}{ccccccccccccc}
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0
\end{array} \right)
\end{array}.
$$

Acknowledgement. The authors thank the referees for providing valuable comments and suggestions.

References

[1] Chung, I., Choi, W., Kim, Y., Lee, M.: The design of conference key distribution system employing a symmetric balanced incomplete block design. Information Processing Letters *81*, 313-318(2002).

[2] Colbourn, C. J.: Projective planes and congestion-free networks. Discrete Applied Mathematics *122*, 117-126(2002).

[3] Eisen, P.A.: Threshold visual cryptography schemes with specified whiteness levels of reconstructed pixels. Designs, Codes and Cryptography *25*(1), 15-61(2002).

[4] Ghafoor, A., Bashkow, T. R., Ghafoor, I.: Bisectional fault-tolerant communication architecture for supercomputer systems. IEEE Transactions on Computers *38*(10), 1425-1446(1989).

[5] Kim, M., Park, J.: New construction of $(2, n)$ visual cryptography for multiple secret sharing. Journal of the Korean Institute of Information Security and Cryptology *10*, 37-47(2003).

[6] Lee, O., Lee, S. , Kim, S., Chung, I.: An efficient load balancing algorithm employing a symmetric balanced incomplete block design. Lecture Notes in Computer Science *2657*, 147-154. Heidelberg:Springer-Verlag 2003.

[7] Naor, M., Shamir, A.: Visual cryptography. Advances in Cryptology-EUROCRYPTO'94, 1-12(1994).

[8] Nugroho, S., Govindarajulu, Z.: Nonparametric tests for random effects in the balanced incomplete block design. Statistics and Probability Letters *56*, 431-437(2002).

[9] Stinson, D. R.: An introduction to combinatorial designs. Preprint, Department of Combinatorics and Optimization, University of Waterloo 1999.

ILUTP_Mem: A Space-Efficient Incomplete LU Preconditioner

Tzu-Yi Chen*

Department of Mathematics and Computer Science, Pomona College,
Claremont CA 91711, USA
tzuyi@cs.pomona.edu

Abstract. When direct methods for solving large, sparse, nonsymmetric systems of linear equations use too much computer memory, users often turn to preconditioned iterative methods. It can be critical in solving such systems to choose a preconditioner which both uses a limited amount of memory, and helps the subsequently applied iterative solver converge more rapidly. This paper describes ILUTP_Mem, an incomplete LU preconditioner that computes an incomplete LU factorization that effectively uses an amount of space specified by the user. The ILUTP_Mem preconditioner is evaluated on a set of matrices from real applications.

Keywords: Sparse nonsymmetric linear systems, iterative methods, incomplete-LU preconditioners

1 Introduction

Direct methods for solving $Ax = b$ first compute the LU factorization of A (ie, $LU = A$) and then solve two triangular systems to find x. Direct methods are robust, but unfortunately can be impractical when computer memory is limited. When direct methods cannot be used, iterative methods become the solvers of choice. Because iterative methods are generally less robust, users often try to improve a method's behavior by applying it to a preconditioned system. Informally, a preconditioner transforms a system into one that is more suited, in some way, for the solver being used. Choosing an effective preconditioner can be critical in solving a system.

The class of incomplete LU (ILU) preconditioners all compute approximate LU factorizations of A (ie, $\hat{L}$ and $\hat{U}$ such that $\hat{L}\hat{U} \approx A$) and use the incomplete

* Much of this work was done while the author was a graduate student at the University of California at Berkeley, where she was supported in part by LLNL Memorandum Agreement No. B504962 under the Department of Energy under DOE Contract No. W-7405-ENG-48, and the National Science Foundation under NSF Cooperative Agreement No. ACI-9619020, and DOE subcontract to Argonne, No. 951322401. The information presented here does not necessarily reflect the position of the policy of the Government and no official endorsement should be inferred.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 20–28, 2004.
© Springer-Verlag Berlin Heidelberg 2004

factors as the preconditioner. In more detail, ILU preconditioners typically compute a complete LU factorization of A, but choose to *drop* (ie, to set their values to 0.0) certain elements along the way. The fewer elements that are dropped, the more effective the preconditioner generally is, but also the more space it uses. Various ways of deciding which elements to drop are discussed in references such as [18]; a popular subclass is that of *value-based* ILU preconditioners, which drop elements whose values are small relative to that of other elements. Examples of value-based ILU heuristics are given in, for example, [1,9,14,15,16].

Value-based heuristics typically require a *drop tolerance*, which is used to determine which elements should be dropped. In general, a *droptol* value of 0.0 means no elements are dropped and a complete LU factorization is computed: the subsequent iterative solver will converge very rapidly, but no memory is saved over a direct solver. When larger *droptol* values are used, the number of elements kept, and hence the memory required, cannot generally be predicted. Therefore, ILU heuristics such as those in [11,12,15,16,19] also, or instead, take a parameter which limits the total number of nonzero elements.

For the remainder of this paper we focus on the ILUTP preconditioner described in [16]. In addition to a *droptol* parameter, ILUTP also takes an *lfil* parameter and uses it as an upper bound on the number of nonzeros in each row of $\hat{L}$ and $\hat{U}$. This method of bounding space requirements encourages users to expect, and to allocate space for, incomplete factors containing a total of $2n \times lfil$ nonzeros. Unfortunately, in practice, we find that even when *lfil* is much less than n, the number of nonzeros in $\hat{L} + \hat{U}$ can be significantly less than $2n \times lfil$.

The plot in Figure 1 shows how few nonzeros the incomplete factors can have. The horizontal axis gives the values used for *lfil* and were chosen to span the range used in, for example, [4,15]. The vertical axis shows the value of $nnz(\hat{L} + \hat{U})/(2n \times lfil)$, where $nnz(\hat{L})$ and $nnz(\hat{U})$ are the number of nonzeros in the incomplete factors computed by ILUTP, for each of the 65 matrices in our test suite. Notice that the ratio can be quite small for these matrices, especially considering that the dimension n ranges from the 1000s to the 100,000s. We used a *droptol* value of 0.0, so that elements are only dropped when there are more than *lfil* elements in a row. More details about the matrices and the testing environment can be found in Section 3.

In other words, Figure 1 shows that on matrices found in practice, even with modest values of *lfil*, considerably fewer than $2n \times lfil$ nonzeros are typically kept in the incomplete factors. Of course, if the subsequently applied iterative solver still finds the solution to the preconditioned system in an acceptable amount of time, this fact would be interesting but perhaps irrelevant. However, if the solver does not, then it seems odd to have computed an unacceptable preconditioner when the memory for a potentially better one was available.

The situation is particularly frustrating since ILU heuristics ask users to supply values for assorted parameters and yet, as shown in Figure 1, may still hide important details about the preconditioner computed. A user who specifies the amount of space available should be able to expect a preconditioner that fully uses the space in order to maximize the chances of solving the system.

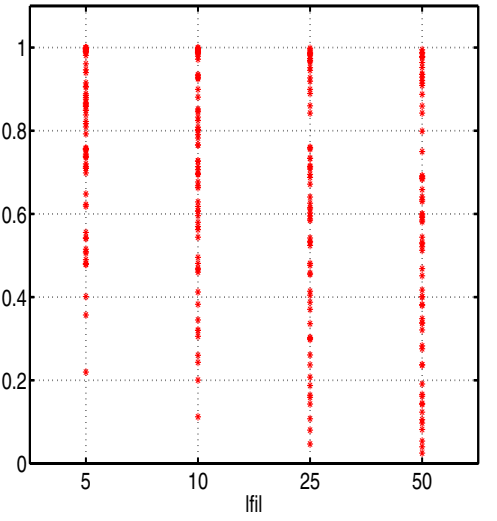


Fig. 1. $nnz(\hat{L}+\hat{U})/(2n \times lfil)$ for different values of *lfil*

In this paper we describe ILUTP_Mem, a value-based ILU heuristic which tries to use the space available for the preconditioner as effectively as possible by adaptively setting the *lfil* value for each row. We compare ILUTP and ILUTP_Mem as preconditioners for restarted GMRES [17] on a test suite of 65 matrices. We conclude with some general recommendations for users as well as some interesting open questions.

2 ILUTP_Mem

The value-based ILUTP_Mem heuristic computes a preconditioner that uses, at most, an amount of space specified by the user. In addition, it also tries to use as much of that space as possible in the hopes of providing as effective a preconditioner as possible. More specifically, the user gives a value for *lfil_nnz*, and ILUTP_Mem uses *lfil_nnz* $\times$ *nnz(A)* as an upper bound on the total number of nonzeros in $\hat{L}+\hat{U}$. As each row of $\hat{L}$ and $\hat{U}$ is computed, the number of nonzeros that can be kept in that row is the total amount of space that is left divided by the number of rows still to be (incompletely) factored. Hence, the first row of $\hat{L}+\hat{U}$ uses at most $1/n$ of the number of nonzeros available for the incomplete factors; the second row at most $1/(n-1)$ of the space remaining after the first row has been stored; and so on for all n rows of $\hat{L}$ and $\hat{U}$.

In addition, ILUTP_Mem was designed to have the same overall structure as ILUTP [16]; this means users can also specify *droptol* and *pivtol* values, which are interpreted just as they are in ILUTP. This means the heuristic returns not

only $\hat{L}$, $\hat{U}$, but also a permutation matrix P because of the potential for partial pivoting. Figure 2 gives pseudocode for the ILUTP_Mem heuristic.

```
(L̂, Û, P) = ILUTP_Mem(A, lfil_nnz, droptol, pivtol)
1     space_left = lfil_nnz · nnz(A)
2     for i ← 1 to n
3         copy A(i,:) into work vector w
4         space_row = space_left/(n − i + 1)
5         lfil = space_row/2
6         for j ← 1 to i − 1
7             if w(j) ≠ 0 and w(j)/Û(j,j) > droptol
8                then w(j : n) = w(j : n) − (w(j)/Û(j,j)) · Û(j, j : n)
9                else w(j) = 0.0
10        L̂(i, 1 : i − 1) = largest lfil elements of w(1 : i − 1).
11        for j ← i to n
12            if w(j) ≤ droptol · ‖A(i,:)‖
13               then w(j) = 0.0
14        Û(i,i) = w(i)
15        lfil = space_row − nnz(L̂(i,:))
16        Û(i, i + 1 : n) = largest lfil − 1 elements of w(i + 1 : n).
17        if max(Û(i, i + 1 : n)) > Û(i,i)/pivtol
18           then pivot by swapping the max and diagonal entries
19                update L̂, Û
20        update P
21        space_left = space_left − nnz(L̂(i,:)) − nnz(Û(i,:))
```

Fig. 2. Pseudocode for ILUTP_Mem

The idea of allowing a different number of nonzeros in each row of the incomplete factors is not new. Other authors have suggested allocating nonzeros in proportion to the number of nonzeros in the original rows of A, or even in proportion to the number of nonzeros in the rows of the complete factors of A [19]. The latter, of course, is not always practical. The strategy used by ILUTP_Mem has the advantage of not requiring specific knowledge about the complete factors, and yet allowing the incomplete factors to become denser in later rows, hence better mimicking the behavior of a complete factorization.

In addition, the *lfil_nnz* parameter given as input is interpreted as a multiple of $nnz(A)$; in other words, ILUTP_Mem guarantees that $nnz(\hat{L}+\hat{U}) \leq lfil_nnz \times nnz(A)$. Since at a minimum $nnz(A)$ space was used to simply store the system A, it seems reasonable to think of the space available for the preconditioner in terms of that number. Other ILU preconditioners, including ILUTP, use bounds that are independent of $nnz(A)$.

3 Methodology

Before comparing ILUTP_Mem and ILUTP, we first describe the framework used for testing and the specific tests done.

We used ILUTP_Mem and ILUTP to precondition a set of 65 test matrices and then tried to solve the preconditioned systems using GMRES(50) [17]. The 65 test matrices were chosen to represent a range of application areas as well as to overlap significantly with the matrices used in other studies such as [4] and [10]. See [3] for a complete list of matrices; most can be downloaded from the University of Florida Sparse Matrix Collection [5]. Similarly we chose GMRES(50) as the iterative solver because studies such as [4] use it as theirs.

Before computing their incomplete factorizations we permuted the rows and columns of the matrices using MC64 [7,8] to maximize the product of the diagonal elements, scaled the matrices so that the new diagonal elements had magnitudes of 1.0, and finally symmetrically permuted them using the ordering generated by COLAMD [6]. We made these choices after conducting extensive tests using assorted fill-reducing orderings as well as various combinations of scalings and settings for MC64. For a more complete description of these tests, see [3]; our results regarding which variant of MC64 to use agree with those in [2] and [7].

Most of the results in this paper use a *droptol* value of 0.0 and a *pivtol* value of 1.0: elements are only dropped for reasons of space, and partial pivoting is used for stability. We arrived at these default values after testing the ILU heuristics with *droptol* values of 0.0, .001, .01, and .1; and *pivtol* values of 0.0, .1, and 1.0. For the *lfil_nnz* parameter we used values ranging from 0 through 5. To compare the results obtained using ILUTP_Mem and ILUTP as preconditioners, we specify *lfil_nnz* for both heuristics (ie, the *lfil* parameter of ILUTP always has the value $lfil_nnz \times nnz(A)/2n$). For more complete results, in particular for values of *droptol* other than 0.0 and of *pivtol* other than 1.0, see [3].

The tests were run on the Berkeley Millennium [13], a cluster of approximately 100 2− and 4−way SMPs running Linux.

4 Analysis

In this section we first use the results of our experiments to show that ILUTP_Mem uses more of the available memory than does ILUTP, and then to show that ILUTP_Mem is a more effective preconditioner.

Figure 3 shows that the ILUTP_Mem heuristic, as expected, uses more of the memory made available by the user. The two plots show the value of $nnz(\hat{L} + \hat{U})/(lfil_nnz \times nnz(A))$ for each of the 65 matrices in our test suite. The factors $\hat{L}$ and $\hat{U}$ are computed by ILUTP in the plot on the left, and by ILUTP_Mem in the plot on the right. We use the default values of 0.0 for *droptol* and 1.0 for *pivtol*.

Clearly the diamonds ($\diamondsuit$) in the plot for ILUTP_Mem are closer to 1.0 than the stars ($*$) in the plot for ILUTP, showing that more of the available space is

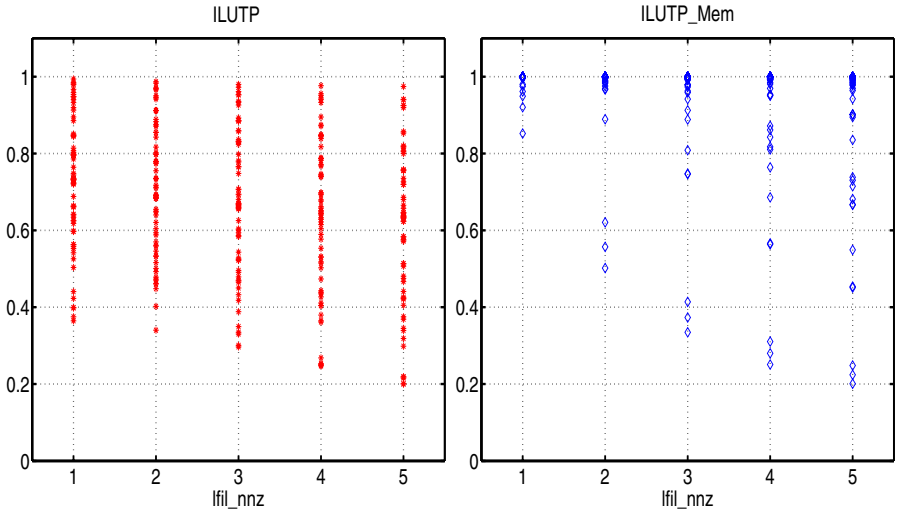


Fig. 3. $\text{nnz}(\hat{L}+\hat{U})$ / $(lfil_nnz \times nnz(A))$ for ILUTP and ILUTP_Mem with different values of *lfil_nnz*

used. In addition, the plot on the right suggests that ILUTP_Mem comes closer to computing a complete factorization when the space for the complete factors is available. Since ILU preconditioners cannot compute anything more accurate than $\hat{L} = L$ and $\hat{U} = U$, once a heuristic computes a complete factorization of a matrix for some value of *lfil_nnz* it should continue to do so as *lfil_nnz* increases, regardless of how much more memory might be available. This explains the three diamonds in the plot for ILUTP_Mem that lie below all others for their values of *lfil_nnz*.

Next we show that the extra memory used by ILUTP_Mem makes a difference in the number of systems solved, which is ultimate test for judging the effectiveness of a preconditioner. Table 1 shows the number of systems that converge for a given value of *lfil_nnz* using ILUTP-preconditioned and ILUTP_Mem-preconditioned GMRES(50). Again, we use a value of 0.0 for *droptol* and of 1.0 for pivtol.

Table 1. Number of systems for which GMRES(50) converged after being preconditioned by ILUTP and ILUTP_Mem

lfil_nnz	1	2	3	4	5
ILUTP	18	23	32	37	41
ILUTP_Mem	19	31	37	40	42

Clearly more systems converge after being preconditioned by ILUTP_Mem than by ILUTP, with the distinction being most marked for *lfil_nnz* values of 2 and 3. Since only 2 of the matrices in the test suite had a value of $nnz(L + U)/nnz(A)$ less than 2, where L and U are the complete factorization of A, this shows that iterative methods can be space-efficient even on some nonsymmetric matrices.

Finally, we show the number of systems that were solved by GMRES(50) for at least one of the twelve combinations of values of *pivtol* and *droptol* that were tested. Table 2 presents similar data to Table 1, except that the numbers now count every preconditioned system that was solved for any combination of the *droptol* and *pivtol* values tried.

Table 2. Number of systems for which GMRES(50) converged with preconditioners computed by ILUTP and ILUTP_Mem

lfil_nnz	1	2	3	4	5
ILUTP	24	31	40	44	44
ILUTP_Mem	29	42	46	49	50

The discrepancies between the data in Table 1 and Table 2 show that the default values of 0.0 for *droptol* and 1.0 for *pivtol* are nowhere near optimal for all systems. For example, even though the results for *lfil_nnz* = 5 in Table 1 suggest that the two preconditioners are almost equally effective, the full results in Table 2 suggests this is not true. Clearly there remains much to be learned from the data. Nevertheless, until we better understand the results, a reasonable ILU preconditioner to try using is ILUTP_Mem with the default values of 0.0 for *droptol*, 1.0 for *pivtol*, as large an *lfil_nnz* as possible, and the orderings and scalings described in Section 3. This preconditioner is particularly appropriate if the space available is a small multiple of $nnz(A)$ and if nothing special is known about the system.

5 Conclusions

In one sense these results are not remarkable: ILUTP_Mem computes a preconditioner with more nonzeros than does ILUTP, therefore the solutions to systems preconditioned using ILUTP_Mem are more easily computed by iterative methods such as GMRES(50). However, this is not the basis on which ILUTP_Mem should be judged. Rather, ILUTP_Mem should be viewed as a heuristic that encourages the user to specify an honest upper bound on the amount of memory they have available for their preconditioner, and to trust the software to compute something reasonable in that space.

One of the advantages of ILUTP_Mem is that it seems to come closer than many other ILU heuristics to computing a complete LU factorization as the

amount of available memory is increased (as specified through the *lfil_nnz* parameter). We are working towards developing an ILU preconditioner that would provably degenerate to a complete LU factorization once the ratio of the amount of memory made available to the amount of memory needed for the complete factors approaches some limit (ideally 1.0, though in practice likely larger). It is not yet clear whether ILUTP_Mem can be modified to achieve this goal.

Acknowledgments. The author would like to thank Jim Demmel for helpful discussions, and the anonymous referees for their comments.

References

1. O. Axelsson and N. Munksgaard. Analysis of incomplete factorizations with fixed storage allocation. In D. Evans, editor, *Preconditioning Methods Theory and Applications*, pages 219–241. Gordon and Breach, 1983.
2. M. Benzi, J. C. Haws, and M. Tuma. Preconditioning highly indefinite and nonsymmetric matrices. *SIAM J. Sci. Comput.*, 22(4):1333–1353, 2000.
3. T.-Y. Chen. *Preconditioning sparse matrices for computing eigenvalues and solving linear systems of equations.* PhD thesis, University of California at Berkeley, December 2001.
4. E. Chow and Y. Saad. Experimental study of ILU preconditioners for indefinite matrices. *J. Comp. and Appl. Math.*, 86:387–414, 1997.
5. T. Davis. University of Florida sparse matrix collection. NA Digest, v.92, n.42, Oct. 16, 1994 and NA Digest, v.96, n.28, Jul. 23, 1996, and NA Digest, v.97, n.23, Jun. 7, 1997. available at: http://www.cise.ufl.edu/∼davis/sparse/.
6. T. A. Davis, J. R. Gilbert, S. I. Larimore, and E. G. Ng. A column approximate minimum degree ordering algorithm. Technical Report TR-00-005, Department of Computer and Information Science and Engineering, University of Florida, October 2000.
7. I. S. Duff and J. Koster. The design and use of algorithms for permuting large entries to the diagonal of sparse matrices. *SIAM J. Matrix Anal. Appl.*, 20(4):889–901, 1999.
8. I. S. Duff and J. Koster. On algorithms for permuting large entries to the diagonal of a sparse matrix. *SIAM J. Matrix Anal. Appl.*, 22(4):973–996, 2001.
9. V. Eijkhout. Overview of iterative linear system solver packages. Lapack working note 141, July 1998.
10. J. R. Gilbert and S. Toledo. An assessment of incomplete-LU preconditioners for nonsymmetric linear systems. *Informatica*, 24:409–425, 2000.
11. M. T. Jones and P. E. Plassmann. An improved incomplete Cholesky factorization. *ACM Trans. on Math. Softw.*, 21(1):5–17, March 1995.
12. C.-J. Lin and J. J. Moré. Incomplete Cholesky factorizations with limited memory. Technical Report MCS-P682-0897, Argonne National Laboratory, August 1997.
13. UC Berkeley Millennium Project. http://www.millennium.berkeley.edu/.
14. N. Munksgaard. Solving sparse symmetric sets of linear equations by preconditioned conjugate gradients. *ACM Trans. on Math. Softw.*, 6:206–219, 1980.
15. Y. Saad. ILUT: A dual threshold incomplete LU factorization. *Numer. Linear Algebra Appl.*, 4:387–402, 1994.

16. Y. Saad. *Iterative methods for sparse linear systems.* PWS publishing company, 1996.
17. Y. Saad and M. H. Schultz. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7(3):856–869, July 1986.
18. Y. Saad and H. A. van der Vorst. Iterative solution of linear systems in the 20th century. *J. of Comp. and Appl. Math.*, 123:1–33, November 2000.
19. M. Suarjana and K. H. Law. A robust incomplete factorization based on value and space constraints. *Int. J. Numer. Meth. Engng.*, 38:1703–1719, 1995.

Optimal Gait Control for a Biped Locomotion Using Genetic Algorithm

Jin Geol Kim[1], SangHo Choi[2], and Ki heon Park[3]

[1] School of Electrical Engineering, Inha University, Inchon, Korea
john@inha.ac.kr
[2] Dept. of Automation Eng., Inha University, Inchon, Korea
tonkex@hanmail.net
[3] School of Electrical and Computer Engineering
Sungkyunkwan University, Suwon, Korea
khpark@skku.ac.kr

Abstract. This paper is concerned with the generation of a balancing trajectory for improving the walking performance. Balancing motion has been determined by solving the second-order differential equation. However, this method caused some difficulties in linearizing and approximating the equation and had some restrictions on using various balancing trajectories. The proposed method in this paper is based on the GA (genetic algorithm) for minimizing the motions of balancing joints, whose trajectories are generated by the fifth-order polynomial interpolation after planning leg trajectories. Real walking experiments are made on the biped robot IWR-Ⅲ, which was developed by Intelligent Robot Control Lab., Inha University. The system has eight degrees of freedom: three pitch joints in each leg and two joints (one roll and one prismatic joint) in the balancing mechanism. Experimental result shows the validity and the applicability of the newly proposed algorithm.

1 Introduction

The role of robots has been more increased nowadays as industrial development is accelerated. Especially, the human-like robot (humanoid) needs to accept the versatile functionality on various working environments [1, 2]. The research fields on the gait control of a biped robot are very wide and diverse, from kinematics and dynamics analysis of the system to the distinct walking and balancing motion by trajectory planning, which is based on human-walking and the interaction with the walking environment.

The biped robot in this paper has two legs and a balancing mechanism composed of a prismatic joint and a revolutionary joint similar to other biped robots [3-5]. This kind of a robot has typically so highly ordered nonlinear-coupled terms that results in difficulties in analysis and control as its mechanical structure. Many studies are going on to avoid this awkwardness and to find the appropriate solution. It is suggested by Takanishi [6] and Lim [7] that the vertical motion of a balancing weight is set to the constant value in order to linearize the dynamic equation.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 29–38, 2004.
© Springer-Verlag Berlin Heidelberg 2004

In this paper, optimal balancing trajectory using the genetic algorithm will be suggested to achieve smoothly stable walking, and the relationship between the spin moment and the balancing trajectory will be investigated.

2 Modeling of Biped Robot

2.1 Mathematical Model and Spin Moment

The dynamic equation of the biped robot IWR-III can be derived from the principle of D'Alembert. Fig. 1 depicts the mass model and its kinematics model.

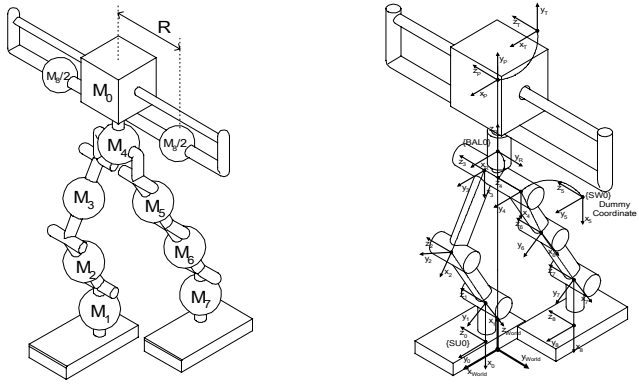


Fig. 1. Mass model and Kinematics model

The biped robot has eight DOFs and one prismatic joint (M_8) and one balancing mass (M_0). The balancing equations induced from its dynamic equation are as follows.

$$M_{Tx} = -\sum_{i=0}^{8} m_i(\ddot{z}_i + G_z)(y_i - y^*) + \sum_{i=0}^{8} m_i(\ddot{y}_i + G_y)z_i = 0 \tag{1}$$

$$M_{Ty} = \sum_{i=0}^{8} m_i(\ddot{z}_i + G_z)(x_i - x^*) - \sum_{i=0}^{8} m_i(\ddot{x}_i + G_x)z_i = 0 \tag{2}$$

$$M_{Tz} = \sum_{i=0}^{8} m_i[(\ddot{x}_i + G_x)(y_i - y^*) - (\ddot{y}_i + G_y)(x_i - x^*)] + m_8 R^2 \tag{3}$$

where, m_i : the mass of i[th] link
x_i, y_i, z_i : the position vector components of i[th] link from the world coordinate
M_{Tx}, M_{Ty}, M_{Tz} : the total moments
G_x, G_y, G_z : the gravitational accelerations
x^*, y^* : the desired frontal and lateral ZMP positions
R : the distance between the origin of balancing joints and the center of gravity (COG) of a balancing weight
θ: the rotational angle of the roll joint

If the spin moment is larger than the friction between foot and ground, walking direction of a robot will be changed by the rotation. As the robot walks faster, the spin moment will be more increased. Therefore, it is desirable to reduce the spin moment by decreasing z-directional moment to improve the walking performance. To do this, it is required to re-design the foot structure with much friction and to add a roll joint on balancing joints. Moreover, a singular point might exist on the center of the balancing joints. Thus, a balancing weight must move through the center of the balancing joints to avoid the singular point during the phase-change.

3 Trajectory Optimization by Genetic Algorithm

3.1 Walking Algorithm

In conventional walking algorithms, trajectories of legs and ZMP (zero moment point) are generated in advance, and then balancing motion trajectory is derived from its dynamic equation [8, 9]. Once the positions of balancing joints are determined by the solution of dynamic equation, entire joint variables and their dynamic properties can be obtained. It is noted that a balancing trajectory depends on the trajectories of legs and ZMP. In order to determine the exact movement of balancing joints, accurate solutions of ordinary differential equations are required. However, the equation has always some mathematical error.

In contrast to this conventional algorithm, GA finds a balancing trajectory with only some via-points. This algorithm guarantees ZMP stability criteria satisfying mechanical restriction without any reconstruction of balancing trajectory. Moreover, by setting the optimization factor to the physically meaningful values such as spin moment, balancing motion and total energy consumption, an additive result can be achieved. Using the genetic algorithm, not by analytical and mathematic approach, we can avoid non-linearity of a biped robot, approximation errors, and mechanical constraint problems. GA sets the ZMP trajectory part as unknown function and detects optimal balancing joints without any analytical and mathematical approaches.

3.2 Balancing Trajectory by Genetic Algorithm

A biped robot should walk maintaining its postural stability. A balancing trajectory must satisfy walking stability criteria for stabilizing the motion of a robot. Two kinds of trajectories are considered for balancing. One is the ZMP trajectory and the other is the balancing joints' trajectory. For a stable walking, ZMP must be located in the supporting foot area. A bit of changes of ZMP can affect the robot's whole stability, so it should carefully be dealt with and planned. ZMP trajectory is related to the balancing joints' trajectory as a coupled form of the second-order non-linear ordinary differential equation. The conventional algorithm acquired the balancing joints' trajectory as a result of solutions of equation with a given ZMP trajectory. However, the proposed algorithm produces a balancing trajectory using GA and now ZMP is just used as an index for stability verification.

Genetic algorithm is a parallel and a global searching algorithm based on the survival of the fittest [10, 11]. Three genetic operators generate the population of the

next generation: reproduction, crossover and mutation. Genetic algorithm can be applied to the various mathematical problems using fitness function with no assumption of continuity or optimization equation. GA is very useful for finding the optimal solution over global search area and has no mathematical limitation for the objective function.

In this paper, the balancing trajectories are generated for the unit step. The fifth-order polynomial interpolation over the time, which minimizes the jerk impact at the via-points, is used for trajectory generation. A biped robot walks continuously with this iterative unit step. The initial-point is predefined in advance, and via-points are set to the origin of the balancing joints in order to satisfy the mechanical constraints of a robot system. And then GA finds the middle via-point at the phase-change time and the last via-point at the end of one-step walking. To find an optimal balancing trajectory, four chromosomes are selected. Two chromosomes have some information for the distances of the balancing prismatic joint at the beginning and end of phase-change. The others have the physical meanings of the rotation angles of the balancing revolutionary joint. The fitness function used in GA is given by

$$f = \sum_{i=t_0}^{t_f} (\theta_i / \theta_a) + \sum_{i=t_0}^{t_f} (d_i / d_a) \tag{4}$$

where θ_a is the full workspace in angle, θ_i is the angle during the one unit time, d_a is the whole movable distance, d_i is the moving distance during the unit time, t_0 and t_f are the initial time and the final time of the one-step walking, respectively.

It takes 3 seconds for a robot to walk one-step. At first, the swing leg takes a step forward for 2 seconds, and then the balancing weight moves from one side to the other opposite side. Thus, phase-change takes place during 1 second. The boundary values are predefined by considering position, velocity, and the acceleration of the balancing joints at the beginning of walk. The velocity and the accelerations at the initial time and final time are set the constant or zero for the simplicity of the algorithm. The searching area for chromosomes is shown in Fig. 2. In this figure, d represents the linear moving distance and θ is the rotating angle of a balancing weight.

Table 1 shows parameters for genetic algorithm. After planning the leg trajectory, the chromosomes are given as boundary values of the balancing joints for the generation of the balancing joints. In these parameters, crossover rate and mutation rate are determined by changing both of them. When changing value of crossover rate and mutation rate, they are increased by 0.1 from 0 to 1. From this simulation result, we can obtain pseudo-optimal value, and value of the fitness function reaches the maximum value around the generations of thirties. In this paper, the number of generation is fixed to 50.

The balancing trajectory having minimum moving distance is determined by genetic algorithm with the initial position. However, all the trajectories are not acceptable because a balancing trajectory must be in the working area and an actual ZMP must be in the stable region. If a balancing trajectory satisfies the above two conditions, the fitness function is calculated. Otherwise, the trajectory will be given up and chromosomes also will be faded away. These procedures are executed repeatedly through the genetic operation of reproduction, crossover and mutation until the number of generation reaches the maximum value. Fig.3 describes the flowchart of two algorithms for comparison.

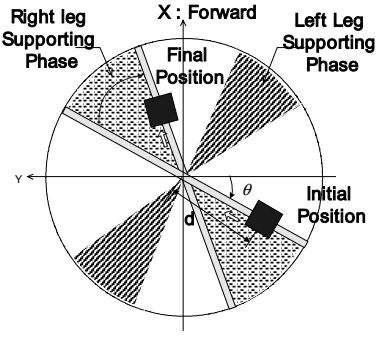


Fig. 2. Searching space.

Table 1. Parameters for genetic algorithm.

Item	Number
number of population	50
number of generation	50
crossover rate (%)	0.3
Mutation rate (%)	0.3
number of genes	4

4 Simulation and Experiments

The system IWR-III has eight AC servomotors, 1/100 of gear ratio of reducer for ankles, 1/60 for the others. The robot is controlled by TMS320C31 DSP controller embedded in the host computer, which analyzes and monitors the overall robot system. A linear sliding guide mechanism is employed for balancing action. 400-Watt actuators are installed with the incremental encoder on two knees and 200-Watt actuators on the other joints. The footprint area is 0.09 m by 0.17 m. Fig. 4 shows the picture and the configuration of the IWR-III biped walking robot, which is 0.685m in height and about 47kg in total weight. Duralumin is used to lighten the robot body.

In order to make biped robot walk, we determine 4 via points each gait. Every walking pattern is finished in 3 second each gait. Until 2 second each gait, a robot supports a single leg state. And then, from 2 second to 3 second, a robot moves balancing joint so as to start next gait. At this time, balancing joint must pass the center of boundary circle to satisfy constraint of kinematics.

The experiment consists of two steps. First, joint angles optimized by GA are calculated through the simulation. Next step is the real-time control of robot with the appropriate PID parameters. Host program is for monitoring the walking motion and control the robot system. Numerical simulator, which is constructed by MATLAB, is made up of the leg trajectory generator, the kinematics-dynamics solvers, the genetic trainer, and the ZMP verifier.

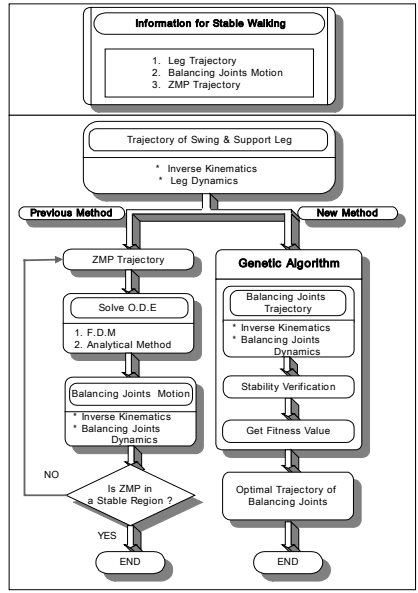


Fig. 3. Flowchart of two algorithms

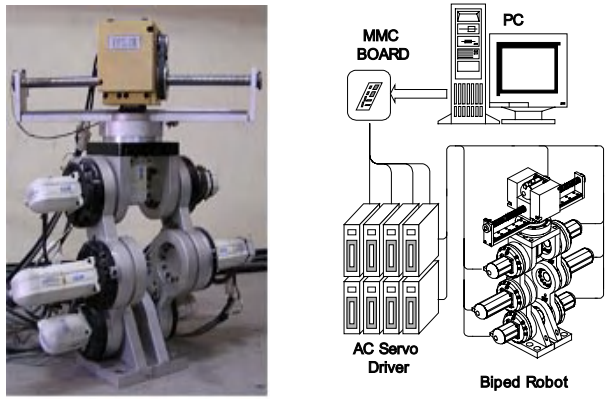
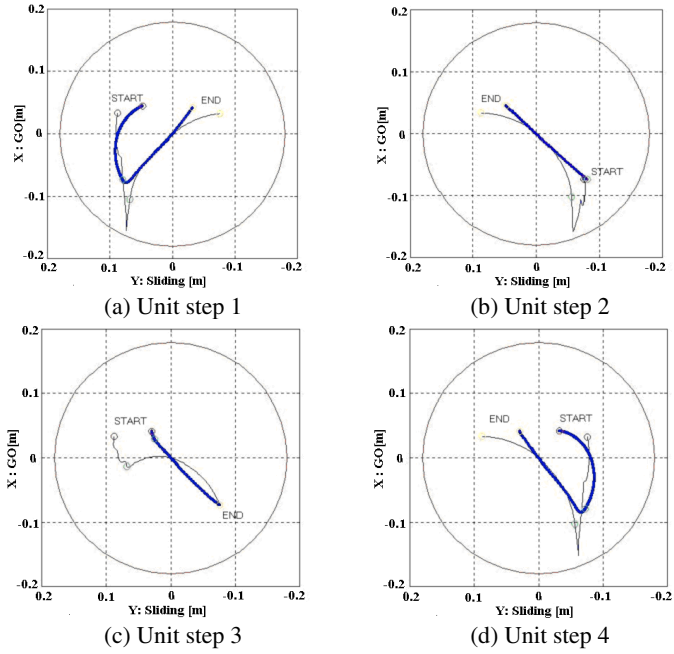


Fig. 4. Picture and configuration of IWR robot system

Fig. 5 shows comparative results of the balancing joints by GA and the conventional method. The thick line represents the optimal trajectory by the proposed genetic algorithm and the thin line denotes the conventional trajectory. Also, the circle represents the workspace of a balancing weight and the y-direction is the walking direction.



(a) Unit step 1 (b) Unit step 2

(c) Unit step 3 (d) Unit step 4

Fig. 5. Balancing trajectories by GA and conventional algorithm.

The final position of the balancing joints in one unit step is set to the starting position of the next unit step. For easy comparison, the initial positions of the balancing joints of two algorithms are set to the same point. At unit step 1, the initial step of the continuous walking, robot starts walking with left swing leg. The balancing joints are located at the right-below position and move to the left-above position during the first phase-change. At unit step 2, the right leg is a leading swing one. The balancing joints move back for a while, and go towards for right-above position across the origin. Unit step 3 is opposite to unit step 2, and unit step 4 is for stopping motion of a robot. Above sequential steps are continuously iterated. The balancing trajectory satisfying the minimum moving distance is determined in each step using GA. The objective function consists of the summation of the linear moving distance and the angular movement.

As expected, the trajectory by GA shows that the balancing motion is much smoother and the moving distance is significantly shortened comparing to the results of the conventional algorithm. It is noted that balancing joints remain on the stable workspace during the overall four-step walking and it moves across the origin point satisfying the mechanical constraint. In the Fig. 6 the ZMP trajectory is shown during the continuous multi-step walking.

The left figure in Fig. 6 shows the ZMP movements from the top view. Rectangles represent the supporting feet. The right figure shows the ZMP movements for frontal and lateral directions during the walk. It is noted that all the ZMPs are on the supporting foot region during 4 steps. Fig. 7 depicts the total spin moments of GA and conventional algorithm.

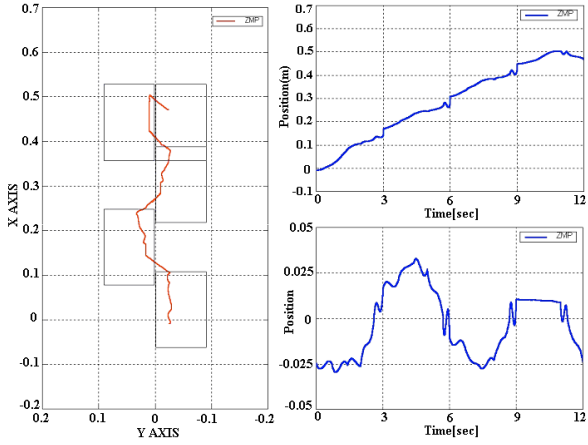
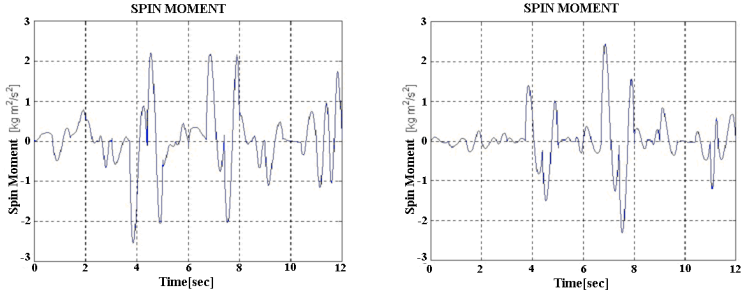


Fig. 6. ZMP trajectory

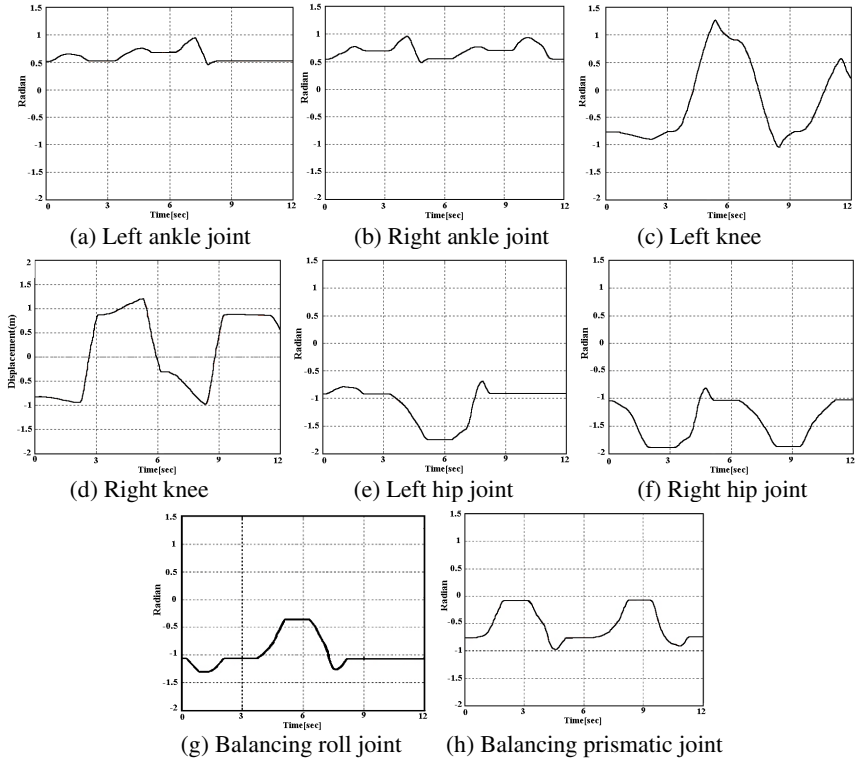


(a) Conventional algorithm (b) Genetic algorithm

Fig. 7. Spin moments of two algorithms

In this figure, Fig.7 (a) illustrates a spin moment by conventional method and Fig. 7 (b) depicts a moment by newly suggested genetic algorithm. It is shown from the Figures 5-7 that optimal balancing trajectory minimizes the moving distance, rotational movement and also curtails spin moments on the whole.

After the inspection of the simulation results, we carried out a real experiment in order to verify them. A controller sends the pulse signal to the servo amplifier per 10 ms and receives the encoder signal. In this experiment, the biped robot walks four gaits, 3 seconds per one gait. The Fig. 8 shows the reference joint angles and the tracking results of the eight servos. Without any radical changes of joint angles, all actuators are well controlled and have little tracking errors.



(a) Left ankle joint (b) Right ankle joint (c) Left knee

(d) Right knee (e) Left hip joint (f) Right hip joint

(g) Balancing roll joint (h) Balancing prismatic joint

Fig. 8. Experimental tracking data of the servo motors

5 Conclusion

Optimal balancing trajectory of a biped robot, minimizing the movement of balancing joints by GA, is proposed and verified through the experiment on the real system IWR-III. A newly optimized trajectory is generated without any consideration of leg trajectories. Using the proposed algorithm, we can acquire the improved walking performance with much reduced spin moments and higher stability. ZMP is used as stabilization index to achieve dynamic walking for a biped robot.

In the near future, advanced new balancing trajectory for z-directional 3-D movement, such as the inverted pendulum type of a humanoid, will be investigated. Various fitness functions for the energy optimization, smooth acceleration and de-acceleration during the walk will be key factors for a smoothly human-like walking. Also, simultaneous propulsion of a trunk and a swing leg for continuous walking must be studied.

Acknowledgement. This work was supported by Grant No. R01-2003-000-10364-0 from Korea Science & Engineering Foundation.

References

1. Kajita, S. et al.: Running Pattern Generation for a Humanoid Robot, Proc. Int. Conf. On Robotics and Automation (2002) 2755-2761
2. Yoshiaki Sakagami, Ryujin Watanabe, Chiaki Aoyama, Shinichi Matsunaga, Nobua Higaki, and Kikuo Fujimura: The intelligent ASIMO:System overview and integration, Proc. Int. Conf. Robotics and Systems (2002) 2478-2483
3. Kawamura S. et al.: Realization of Biped Locomotion by Motion Pattern Learning, Journal of Robot Society of Japan vol.3 No.3 (1985) 177-180
4. Arimoto S., Kawamura S., and Miyazaki F.: Can Mechanical Robots Learn by Themselves, Robotics Research, MIT Press (1985) 127-134
5. Ching-Long Shih: Analysis of the Dynamics of a Biped Robot with Seven Degrees of Freedom, IEEE International Conference on Robotics and Automation (1996) 3008-3013
6. Atsuo Takanish: Robot Biped Walking Stabilized with Trunk Motion, Robots and Biological Systems: Towards a new Bionics, Spring-Verlag (1989)
7. S.H.Lim and J.G.Kim: Adaptive Gait Algorithm for IWR Biped Robot, Proceedings of the International Conference on Power Electronics and Drive Systems, vol.1 (1995) 438-442.
8. Q.Li, A.Takanishi, and I.Kato: Learning Control of Compensative Trunk Motion for Biped Walking Robot based on ZMP Stability Criterion, Proceedings of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems, Raleigh, USA, Jul.7-10 (1992) 597-603
9. Yamaguchi, A.Takanishi, and I.Kato: Development of a Biped Walking Robot Compensating for Three-Axis Moment by Trunk motion, Proceedings of the 1993 IEEE International Conference on Intelligent Robots and Systems, vol.1, Yokohama, Japan, Jul.26-30 (1993) 561-566
10. M.Y.Cheng and C.S.Lin: Genetic Algorithm for Control Design of Biped Locomotion, Journal of Robotic Systems (1997) 365-373
11. D.E. Goldberg: Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, Massachusetts (1989)

A Bayes Algorithm for the Multitask Pattern Recognition Problem – Direct and Decomposed Independent Approaches

Edward Puchala

Wroclaw University of Technology, Chair of Systems and Computer Networks,
Wybrzeze Wyspianskiego 27, 50-370 Wroclaw, Poland
puchala@zssk.pwr.wroc.pl

Abstract. The paper presents algorithms of the multitask recognition for the direct approach and for the decomposed independent approach. Both algorithms are presented in the even of full probabilistic information. Algorithms with full probabilistic information were working on basis of Bayes decision theory. Full probabilistic information in a pattern recognition task, denotes the knowledge of the classes probabilities and the class-conditional probability density functions. Optimal algorithms for the selected loss function will be presented.

1 Introduction

The classical pattern recognition problem is concerned with the assignment of a given pattern to one and only one class from a given set of classes [3]. Multitask classification problem refers to a situation in which an object undergoes several classification tasks. Each task denotes recognition from a different point of view and with respect to different set of classes. For example, such a situation is typical for compound medical decision problems where the first classification denotes the answer to the question about the kind of disease; the next task states recognition of the stadium of disease, the third one determines the kind of therapy, etc. Let us consider the non-Hodgkin lymphoma as a common dilemma in haematology practice. For this medical problem we can utilise the multitask classification (this is motivated by the structure of the decision process), which leads to the following scheme. In the first task of recognition, we arrive at a decision i_1 about the lymphoma type. After the type of lymphoma has been determined, it is essential for diagnosis and therapy to recognize its stage. The values of decision i_2 denote the first, the second, the third and the fourth stage of lymphoma development, respectively. Apart from that, each stage of lymphoma may assume two forms. Which of such forms occurs is determined by decision i_3. If i_3=1, then lymphoma assumes the form A (there are no additional symptoms). For i_3=2, lymphoma takes on form B (there are other symptoms, as well).

Decisions i_4 determines therapy, which is one of the known schemes of treatment (e.g. CHOP, BCVP, COMBA, MEVA, COP-BLAM-I). A therapy (scheme of treatment) cannot be used in its original form in every case. Because of the side

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 39–45, 2004.
© Springer-Verlag Berlin Heidelberg 2004

effects of cytostatic treatment it is necessary to modify such a scheme. Decision about modification is i_5. In the present paper I have focused my attention on the concept of multitask pattern recognition. In particular, so-called direct approach (DA) and decomposed independent approach (DIA) for the problem solution will be taken into consideration. DIA may be worked practically in the computer network [1].

2 Direct Approach (DA) to the Multitask Pattern Recognition Algorithm

Let us consider *N*-task pattern recognition problem. We shall assume that the vector of features $x_k \in X_k$ and the class number $j_k \in M_k$ for the *k*-th recognition task of the pattern being recognized are observed values of random variables $\mathbf{x_k}$ and $\mathbf{j_k}$, respectively [6]. When *a priori* probabilities of the whole random vector $\mathbf{j}=(\mathbf{j_1},\mathbf{j_2},...,\mathbf{j_N})$ denote as $P(\mathbf{j}=j)=p(j)=p(j_1,j_2,..j_N)$ and class-conditional probability density functions of $\mathbf{x}=(\mathbf{x_1},\mathbf{x_2},...,\mathbf{x_N})$ denote as $f(x_1,x_2,..x_N/j_1,j_2,...,j_N)$ are known then we can derive the optimal Bayes recognition algorithm minimizing the risk function [7], [5]:

$$R = \mathbf{E}\, L(\mathbf{i}, \mathbf{j}) \tag{1}$$

i.e. expected value of the loss incurred if a pattern from the classes $j = (j_1, j_2, ..., j_N)$ is assigned to the classes $i = (i_1, i_2, ..., i_N)$.

In the case of multitask classification we can define the action of recognizer, which leads to so-called direct approach. [4].

In that instance, classification is a single action. The object is classified to the classes $i = (i_1, i_2, ..., i_N)$ on the basis of full features vector $x = (x_1, x_2, ..., x_N)$ simultaneously. That we can see below (Figure 1).

Let $\Psi(x)$ denotes direct pattern recognition algorithm:

$$\Psi(x) = \Psi(x_1, x_2, ..., x_N) = (i_1, i_2, ..., i_N) \;\; x_k \in X_k, \; i_k \in M_k \tag{2}$$

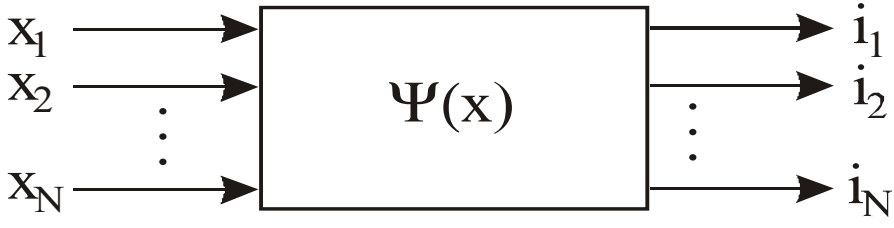


Fig. 1. Block scheme of the direct multitask pattern recognition algorithm.

The minimization of the risk function R leads to the optimal algorithm Ψ^*.

$$R[\Psi(x)] = E\{L[(i_1, i_2, ..., i_N), (j_1, j_2, ..., j_N)]\} \tag{3}$$

$$R(\Psi^*) = \min_{\Psi} R(\Psi) \tag{4}$$

Symbol L denotes the loss function.
Average risk (3) expresses formula:

$$R(\Psi) = \int_X \{ \sum_{j_1 \in M_1} \sum_{j_2 \in M_2} ... \sum_{j_N \in M_N} L[(i_1, i_2, ...i_N), (j_1, j_2, ..., j_N)] * \\ * \ p(j_1, j_2, ...j_N / x)\} f(x)\, dx \tag{5}$$

where:

$$p(j_1, j_2, ...j_N / x)\} = \frac{p(j_1, j_2, ...j_N)\, f(x / j_1, j_2, ...j_N)}{f(x)} \tag{6}$$

denotes *a'posteriori* probability for the set of classes $j_1, j_2, ..., j_N$. As we can easily show the formula:

$$r(i_1, i_2 ...i_N, x) = E[L(i_1, i_2, ..., i_N), (j_1, j_2, ...j_N) / x] = \\ = \sum_{j_1 \in M_1} \sum_{j_2 \in M_2} ... \sum_{j_N \in M_N} L[(i_1, i_2, ..., i_N), (j_1, j_2, ..., j_N)] \times p(j_1, j_2, ..., j_N / x) \tag{7}$$

presents average conditional risk. Hence, the Bayes algorithm for multitask pattern recognition for direct approach may be derived. As we can see, it is result of optimization problem (4) solution. Thus, we have obtained optimal algorithm (8),(9)

$$\Psi^*(x) = (i_1, i_2, ..., i_N) \text{ if}$$

$$r(i_1, i_2, ..., i_N, x) = \min_{i_1', i_2', ...i_N'} r(i_1', i_2', ..., i_N', x) \tag{8}$$

$$\Psi^*(x) = (i_1, i_2, ..., i_N) \text{ if}$$

$$\sum_{j_1 \in M_1} \sum_{j_2 \in M_2} ... \sum_{j_N \in M_N} L[(i_1, i_2, ..., i_N), (j_1, j_2, ..., j_N)] \times p(j_1, j_2, ..., j_N) \times \\ \times f(x / j_1, j_2, ..., j_N) = \\ = \min_{i_1', i_2', ..., i_N'} \sum_{j_1 \in M_1} \sum_{j_2 \in M_2} ... \sum_{j_N \in M_N} L[(i_1', i_2', .., i_N'), (j_1, j_2, ..., j_N) \times \\ \times \ p(j_1, j_2, ...j_N) \times f(x / j_1, j_2, ...j_N) \tag{9}$$

Let us consider characteristic form of loss function L. Value of this function depends on number of misclassification decisions:

$$L[(i_1, i_2, ..., i_N), (j_1, j_2, ..., j_N)] = n \tag{10}$$

where n denotes number of pairs (algorithm's decision i_k and real class j_k) for witch $i_k = j_k$. In this case, average conditional risk has the following form:

$$r(i_1, i_2, ..., i_N, x) = N - [p(i_1 / x) + p(i_2 / x) + ... + p(i_N / x)] \tag{11}$$

Because number of tasks N is constant for each practical problem and we are looking for minimum of average conditional risk, then optimal multitask pattern recognition algorithm for so called direct approach will be allowed to write by formula (12)

$$\Psi^*(x) = (i_1, i_2, ..., i_N) \text{ if}$$
$$\sum_{k=1}^{N} p(i_k / x) = \max_{i_1', i_2', ..., i_N'} \sum_{k=1}^{N} p(i_k' / x) \tag{12}$$

The average risk function, for the loss function L (10), is the sum of the incorrect classification probabilities in individual tasks:

$$R[\Psi] = \sum_{n=1}^{N} P_e(n) = \sum_{n=1}^{N} [1 - P_c(n)]$$
$$P_c(n) = \sum_{j_1 \in M_1} \sum_{j_2 \in M_2} ... \sum_{j_N \in M_N} q(j_n / j_1, j_2, ..., j_N) \times p(j_1, j_2, ..., j_N) \tag{13}$$

where $q(j_n / j_1, j_2, ..., j_N)$ is the probability of correct classification for object from classes $(j_1, j_2, ..., j_N)$ in n-th task:

$$q(j_n / j_1, j_2, ..., j_N) =$$
$$= \sum_{i_1 \in M_1} ... \sum_{i_{n-1} \in M_{n-1}} \sum_{i_{n+1} \in M_{n+1}} ... \sum_{i_N \in M_N} \int_{D_x^{(i_1, ..., i_N)}} f(x / j_1, ..., j_N) dx \tag{14}$$

$D_x^{(i_1, ..., i_N)}$ - decision area for algorithm $\Psi(x)$.

3 Decomposed Independent Approach (DIA) to the Multitask Pattern Recognition Algorithm

Let us consider second algorithm in which we have N independent tasks, what we can see on the figure 2. Now, we deal with N independent recognition algorithms for each of tasks:

$$\begin{aligned} \Psi_1(x_1) &= i_1 \\ \Psi_2(x_2) &= i_2 \\ &\cdots \\ \Psi_N(x_N) &= i_N \end{aligned} \tag{15}$$

Our problem consists in determination of N optimal (in Bayes sense) pattern recognition algorithms. Let us consider characteristic form (0-1) of loss function L.

$$L[(i_n, j_n)] = \begin{cases} 1 \ \ if \ \ i_n \neq j_n \\ 0 \ \ if \ \ i_n = j_{nc} \end{cases} \tag{16}$$

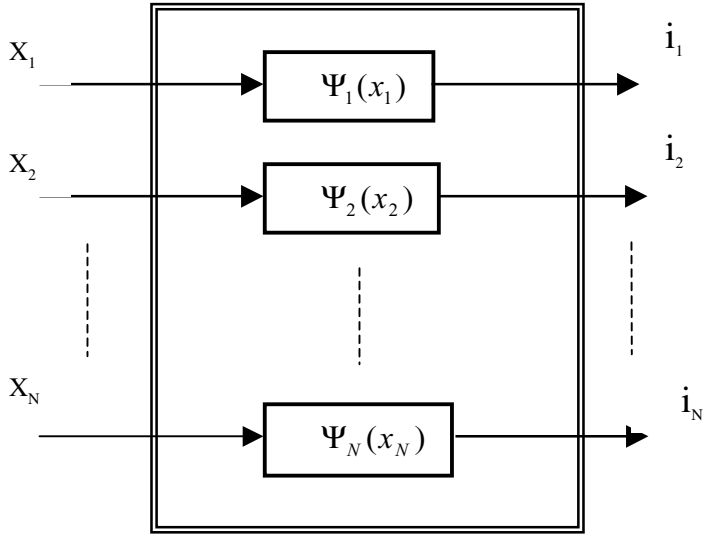


Fig. 2. Block scheme of the decomposed independent multitask pattern recognition algorithm

In this case we obtain the following optimal algorithm for n-th task:

$$\begin{aligned} &\Psi_n(x_n) = i_n, \ \ if \\ &\sum_{j_n} L_n(i_n, j_n) \times p(j_n / x_n) = \min_{k_n} \sum_{j_n} L(k_n, j_n) \times p(k_n / x_n); \end{aligned} \tag{17}$$

where:

$$p(j_n / x_n) = \frac{f(x_n / j_n) \times p(j_n)}{\sum_{j_n} f(x_n / j_n) \times p(j_n)} \tag{18}$$

The superiority the multitask algorithm in direct and decomposed version over the classical pattern recognition one demonstrates the effectiveness of this concept in such multitask classification problems for which the decomposition is necessary from the functional or computational point of view (e.g. in medical diagnosis). Direct approach to multitask recognition algorithms gives better results then decomposed approach because such algorithms take into consideration correlation between individual classification problems. Results of the experiments are shown on the chart bellow (Figure 3.).

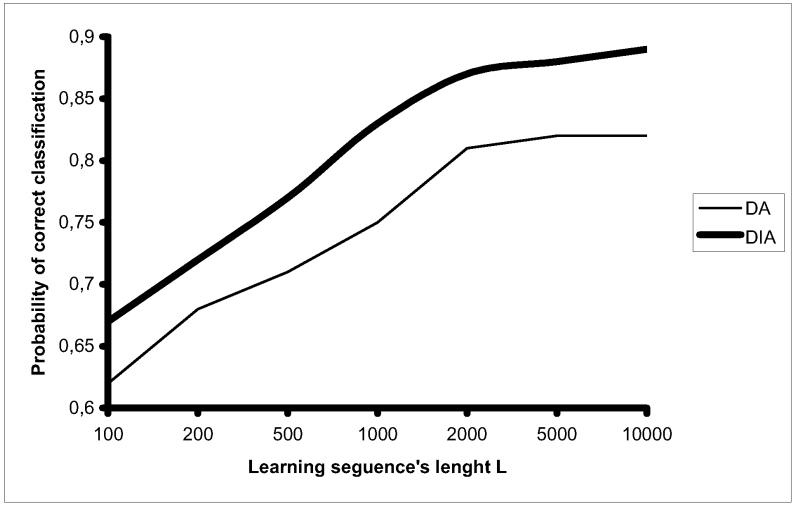


Fig. 3. Probability of multitask pattern recognition algorithms correct classification correspondence to length of learning sequence S_L for direct approach (DA) and for decomposed independent approach (DIA).

Of course, in the formulas (12), (13), (14) for direct approach and in the formulas (17), (18) for decomposed independent approach, estimators of probabilities *p, q* and density functions must be applied. These estimators were obtained on the base data which are in so called learning sequence S_L.

$$S_L = \left\{ \left(x^1, j^1\right), \left(x^2, j^2\right), \ldots, \left(x^L, j^L\right) \right\}$$

where:

L – length of learning sequence,

$x^k = (x_1^k, x_2^k, \ldots, x_N^k)$ - features vectors for tasks 1, 2,…,N,

$j^k = (j_1^k, j_2^k, \ldots, j_N^k)$ - class numbers for tasks 1, 2,…, N

Acknowledgement. The work presented in this paper is a part of the project realized in the University of Applied Sciences in Legnica (Poland)

References

1. Gola M., Kasprzak A.,: The Two-Criteria Topological Design Problem in WAN with Delay Constraint: An Algorithm and Computational Results, Lecture Notes in Computer Science, vol. 2667, 2003, pp 180-189
2. Wozniak M.,: Proposition of the quality measure for the probabilistic decision support system, Lecture Notes in Computer Science, Lecture Notes in Artificial Intelligence, vol. 2718, 2003, pp.686-691
3. Puchala, E., Kurzynski, M.,: A branch-and-bound algorithm for optimization of multiperspective classifier. Proceedings of the 12th IAPR, Jerusalem, Israel, (1994) 235-239
4. Kurzynski, M., Puchala, E., :Algorithms of the multiperspective recognition. Proc. of the 11th Int. Conf. on Pattern Recognition, Hague (1992)
5. Duda, R., Hart, P.,: Pattern classification and scene analysis. John Wiley & Sons, New York (1973)
6. Fukunaga, K., : Introduction to Statistical Pattern Recognition, Academic Press, New York (1972)
7. Parzen, E.,: On estimation of a probability density function and mode. Ann. Math. Statist., (1962) Vol.33, 1065-1076

Energy Efficient Routing with Power Management to Increase Network Lifetime in Sensor Networks*

Hyung-Wook Yoon, Bo-Hyeong Lee, Tae-Jin Lee, and Min Young Chung

School of Information and Communication Engineering
Sungkyunkwan University, Suwon, KOREA
{hwyoon, shaak, tjlee, mychung}@ece.skku.ac.kr

Abstract. A sensor network consists of many low-cost, low-power, and multi-functional sensor nodes. One of the most important issues in sensor networks is to increase network lifetime, and there have been researches on the problem. In this paper, we propose a routing mechanism to prolong network lifetime, in which each node adjusts its transmission power to send data to its neighbors. We model the energy efficient routing with power control and present an algorithm to obtain the optimal flow solution for maximum network lifetime. Then we show that our mechanism can save power consumption and increase network lifetime than the mechanism without power management.

1 Introduction

A sensor network consists of low-cost, low-power and multi-functional sensor nodes. Sensor nodes have power supply unit, sensing components to gather information, data processing unit and communication unit to transmit and receive data. Sensor networks can be used in wide variety of applications. For example, a wireless sensor network can be deployed in the area where chemical or biological attack was tried in order to identify contamination by chemical/biological agents. It can be used in a disaster area to collect information about trapped survivors.

Usually sensor nodes are battery-operated, so it is important to minimize the power consumption of a sensor network. In this context, one of the most important issues in sensor networks is to increase network lifetime, the time until the battery of any node drains-out for the first time. If one or more sensor nodes monitoring a region are out of service due to battery outage, useful information may not be collected in the region. So many researches have been focusing on increasing network lifetime in sensor networks.

There have been approaches to increase network lifetime by maintaining only minimal set of working nodes and turning off the others [4],[5]. That is each node assesses its connectivity and determines the state of the node considering the neighbor environment. Other papers propose some energy-efficient Medium Access Control (MAC) or routing protocols for sensor networks [6],[7].

In this paper, we propose an energy-efficient routing mechanism to increase network lifetime, in which nodes are allowed to adjust their transmission power and thus

* This paper was partially supported by BK21 program. The corresponding author is Tae-Jin Lee.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 46–55, 2004.
© Springer-Verlag Berlin Heidelberg 2004

their communication range so that each node in a network can reduce the power consumption to send information and prolong network lifetime. We formulate on optimal problem, and propose an algorithm to maximize network lifetime. We present how much amount of network lifetime is improved by the proposed routing method adopting power management via modeling of networks and performance evaluation.

This paper is organized as follows. In section 2, related works are discussed. In Section 3, we present a power management mechanism, where a sensor node can change transmission power, and derive network lifetime given information to send. In Section 4, we present a proposed routing algorithm. Then the proposed method is analyzed and compared with the other mechanism in Section 5. Finally we summarize the main results and discuss future research directions in Section 6.

2 Related Works

Some researches have tried to increase network lifetime by deceasing energy consumption at each node [4],[5],[7]. They propose power control mechanisms or MAC protocols to save unnecessary power consumption by each node based on the observation of local environment. The method to prolong network lifetime by efficient routing has been researched in Mobile Ad-hoc NETworks (MANETs) and applied to sensor networks. In the method, the route with the minimum total power consumption is selected. This routing method is called Minimum Total Transmission Power Routing (MTPR) [11].

Total transmission power along a route is an important metric because it concerns the lifetime. Although MTPR can reduce the total power consumption of the entire network, it is not directly associated with the lifetime of each node. If some minimum total transmission power routes traverse a specific node, the battery of this node will be exhausted quickly, leading to broken paths. Therefore, the remaining battery capacity of each node may be a more accurate metric for lifetime of each node. So minimum Battery Cost Routing (MBCR) [12] is proposed, in which remaining battery capacity of each node is considered as cost for routing decision.

The Min-Max Battery Cost Routing (MMBCR) mechanism makes the energy of each node to be used rather fairly by trying to avoid the route having the node with the least battery capacity [12]. And in Conditional MMBCR (CMMBCR) [6], a route with the minimum total transmission power among the candidate routes through which all the nodes have sufficient remaining battery capacity above a certain threshold is selected.

There has been proposed energy-saving routing and formulated it as an optimization problem in which the objective is to maximize the network lifetime [2]. An extended model is introduced, in which the nodes have limited bandwidth as well as limited battery [1].

3 Modeling of Sensor Networks

Let N and L denote the set of sensor nodes and the set of directional links connecting nodes, respectively. Thus we model a sensor network by a graph $G = (N, L)$. The type of a node could be a general sensor node, a sink node, and the central unit. A sensor node plays a role to transmit its own sensed information to neighbor nodes and to relay

information received from other nodes. The information of a sensor node is eventually passed to a sink node (the set of sink nodes is denoted by S) and the information collected in a sink node is transmitted to the central unit (the set of central units is denoted by D) through wired or wireless medium. If the information of each node is concurrent and multi-commodity, all the information generated in a sensor network is transmitted to the central unit via sink nodes. A node j that is within the transmission range from node i is assumed to be connected to node i by a directional link (i, j). The collection of nodes connected to node i by directional links is denoted by $Z(i)$.

3.1 Model without Power Management

Let $F_{i,j}$ be the average information flow from node i to node j that is within the transmission range from node i. We define flow $f_{i,j}$ ($0 \leq f_{i,j} \leq 1$) on link $(i, j) \in L$ as the ratio between $F_{i,j}$ and the maximal possible flow, F_{max}, on any link connecting two nodes.

$$f_{i,j} = \frac{F_{i,j}}{F_{max}}.$$

The data traffic generation rate at node i is defined by q_i, and the ratio r_i ($0 \leq r_i < 1$) between q_i and the maximal possible flow on any link connecting two nodes is

$$r_i = \frac{q_i}{F_{max}}.$$

Each node has an energy source, e.g., battery. We assume that node i has an initial energy level E_i and the transmission energy e_0 is required at node i to transmit an information unit. Then lifetime T_i of node i and lifetime T of the network can be defined as follows[1]:

Definition 1. *The lifetime of node i under given flow to be transmitted is the time until the battery of the node drains out:*

$$T_i = \frac{E_i}{\sum_{j \in Z(i)} e_0 f_{i,j}}.$$

Definition 2. *The lifetime of the network under given flow to be transmitted is the time until the battery of any node drains out for the first time, namely the minimum lifetime among all nodes:*

$$T = \min_{i \in N} T_i.$$

Let $\overline{f_{ij}}$ be the amount of information transmitted from node i to node j until T, i.e.,

$$\overline{f_{i,j}} = f_{i,j} T. \tag{1}$$

The link flow then should satisfy the following conditions in the network.

$$\begin{aligned}
&\overline{f_{i,j}} \geq 0, && {}^{\forall}(i,j) \in L \\
&\sum_{k \in Z(i)} \overline{f_{k,i}} + r_i \cdot T = \sum_{j \in Z(i)} \overline{f_{i,j}}, && {}^{\forall} i \in N - \{S, D\} \\
&\sum_{k \in Z(i)} \overline{f_{k,i}} = \sum_{j \in Z(i)} \overline{f_{i,j}}, && {}^{\forall} i \in S \\
&\sum_{k \in Z(i)} \overline{f_{k,i}} + \sum_{j \in Z(i)} \overline{f_{i,j}} \leq T, && {}^{\forall} i \in N - \{S, D\}.
\end{aligned}$$

3.2 Proposed Model with Power Management

In order to maximize network lifetime we propose that sensor nodes employ power management. Assuming TX power can be adjusted according to the distance between two nodes, the power consumption to transmit an information unit from node i to node j can be expressed as

$$e_{i,j} = \left(\frac{d_{i,j}}{d_0}\right)^{\alpha} \cdot e_0,$$

where d_0 is the maximum TX range of a node with its maximum TX power, $d_{i,j}$ is the distance between node i and j, and α is a loss constant in the range between 2 and 4. Since energy is infinite at a sink node, a sink node does not affect network lifetime. Thus lifetime T_i of node i and lifetime T of the network can be written as

$$\begin{aligned}
T_i &= \frac{E_i}{\sum_{j \in Z(i)} e_{i,j} f_{i,j}} = \frac{E_i}{\sum_{j \in Z(i)} \left(\frac{d_{i,j}}{d_0}\right)^{\alpha} e_0 f_{i,j}} \\
&= \frac{E_i}{e_0 \sum_{j \in Z(i)} \left(\frac{d_{i,j}}{d_0}\right)^{\alpha} f_{i,j}},
\end{aligned} \tag{2}$$

and

$$\begin{aligned}
T &= \min_{i \in N} T_i \\
&= \min_{i \in N} \left(\frac{E_i}{e_0 \sum_{j \in Z(i)} \left(\frac{d_{i,j}}{d_0}\right)^{\alpha} f_{i,j}} \right).
\end{aligned} \tag{3}$$

We assume that flows are concurrent and feasible. The transmission range of node i should be determined as the farthest distance from node i to any node k connected with node i. Then

$$T_i = \frac{E_i}{e_0 \sum_{j \in Z(i)} \left(\max_{k \in Z(i)} \frac{d_{i,k}}{d_0}\right)^{\alpha} f_{i,j}},$$

and

$$T = \min_{i \in N} \left(\frac{E_i}{e_0 \sum_{j \in Z(i)} \left(\max_{k \in Z(i)} \frac{d_{i,k}}{d_0} \right)^{\alpha} f_{i,j}} \right)$$

$$= \min_{i \in N} \left(\frac{E_i}{e_0 \left(\max_{k \in Z(i)} \frac{d_{i,k}}{d_0} \right)^{\alpha} \sum_{j \in Z(i)} f_{i,j}} \right). \tag{4}$$

And the amount of the flow on the links connected to node i in the network is determined by the energy of the node and the distances between the neighbor nodes and node i from (1) and (4). Thus

$$\sum_{j \in Z(i)} \overline{f_{i,j}} \leq \frac{E_i}{e_0 \left(\max_{k \in Z(i)} \frac{d_{i,k}}{d_0} \right)^{\alpha}}, \quad i \in N - \{S, D\}.$$

The node capacity C_i that the amount of traffic that node i transmits during a unit time is denoted as

$$C_i = \frac{E_i}{e_0 \left(\max_{j \in Z(i)} \frac{d_{i,j}}{d_0} \right)^{\alpha}} \cdot \frac{1}{T_i}, \quad i \in N - \{S, D\} \tag{5}$$

$$C_i \leq \frac{E_i}{e_0 \left(\max_{j \in Z(i)} \frac{d_{i,j}}{d_0} \right)^{\alpha}} \cdot \frac{1}{T}, \quad i \in N - \{S, D\}. \tag{6}$$

Then the flow that each node transmits can be determined from the node capacity. If the generated flow at each node is concurrent and feasible, then the overall information that is generated in the sensor network is eventually transmitted to the central unit via sink nodes.

4 Proposed Routing Algorithm

In order to obtain the optimal feasible flows that are concurrent, the maxflow algorithm is used [8]. In the max flow algorithm, node i is divided into two subnodes (i_i and i_o) connected by an internal link (see Fig. 1). If node i generates information and transmits it to node j, then we assume that it is generated at subnode i_i and is transmitted to subnode i_o, j_i and j_o. So every directional link (i, j) connecting node i and j should be replaced by a directional link (i_o, j_i). Accordingly, the capacity of the internal link (C_{i_i,i_o}) is defined as:

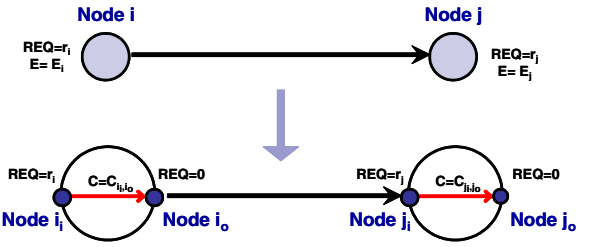


Fig. 1. Transformation of a node-capacitated network to a link-capacitated network.

$$C_{i_i,i_o} = \min_{i \in N} \left(1, \frac{E_i}{e_0 \cdot \left(\max_{j \in Z(i)} \frac{d_{i,j}}{d_0} \right)^{\alpha} \cdot T_i} \right).$$

Then we transform a node-capacitated network to a link-capacitated network as shown in Fig. 1. All information generated by a sensor node is transmitted via links under capacity constraints. In other words, while network is not partitioned, all information generated in the sensor network can be transmitted to the central unit. A set of flows satisfying this condition is called feasible.

We propose an algorithm in order to find feasible flow on each link and to maximize network lifetime (see Fig. 2). The proposed algorithm consists of two parts. First, link capacity C_i is computed from (5) and (6). Then we use the maxflow algorithm to determine the flows along the links. Next, the maximum feasible time is obtained by binary search. The algorithm terminates when the difference between the feasible and non-feasible network lifetime is within a tolerance.

5 Performance Evaluation

In order to evaluate performance a network topology shown in Fig. 3 is considered as in [1]. Distances between nodes are set to random constants to employ power management at nodes. Initial T_{max} is set to 100. Path loss factor α and maximum transmission range d_0 are assumed to be 2 and 10m, respectively.

The flow and residual energy of the previous method [1] to transmit information with the maximum transmission power without power management is shown in Table 1 and 2. The network lifetime T is 7.69s. And the flow and residual energy of the proposed method to transmit information as far as the farthest node among the set $Z(i)$ with the appropriate TX power is given in Table 3 and 4. The network lifetime T becomes 11.33s, which is 47% increase compared to that of the previous method.

So our proposed method is shown to improve network lifetime by adjusting transmission range via power management of sensor nodes. Assuming we have the same amount of information to transmit, consumed energy at nodes in our method is less than that in the method without power control, resulting in longer network lifetime. In other words, the amount of information that each node can send during network lifetime is

```
// R_i : Remaining information to send at node i after the maxflow algorithm
// T : Feasible time (network life time)
// Feasible_Time/nonFeasible_Time : Feasible/non-feasible time in iterations

  Transform the node-capacitated network to the link-capacitated network
  Initialize Tolerance, T_max, Feasible_Time and nonFeasible_Time.

  Set d_i^max = ( max_{k∈Z(i)} d_{i,k}/d_0)^α, i ∈ N − {S, D}, and t = T_max

  While | Feasible_Time - nonFeasible_Time | < Tolerance
        {
        C_i = (E_i/(e_0 · d_i^max))/t, i ∈ N − {S, D}
        Find the flows using the maxflow algorithm and R_i
        // Check if t is feasible
        if R_i = 0, i ∈ N − {D},
              Update Feasible_Time = t
        else
              Update non_Feasible Time = t
        Update t using binary search
        }

  Obtain network lifetime T = t

  Obtain the feasible flows on the links with lifetime T
```

Fig. 2. The proposed algorithm to obtain the flows for maximizing network lifetime and flow.

Table 1. Flow at each link without power management.

$f_{N1,N2}$	$f_{N3,N1}$	$f_{N1,N5}$	$f_{N2,S1}$	$f_{N3,N4}$
0.35	0.25	0.30	0.65	0.15
$f_{N5,S2}$	$f_{N4,S1}$	$f_{N4,S2}$	$f_{N2,N4}$	$f_{S1,D} + f_{S2,D}$
0.70	0.35	0.30	0	2.0

Table 2. Consumed energy without power management during optimal feasible time.

Node	Initial Energy	Energy Consumption	Residual Energy
N1	5	5	0
N2	5	5	0
N3	10	3.07	6.93
N4	5	5	0
N5	10	5.38	4.62

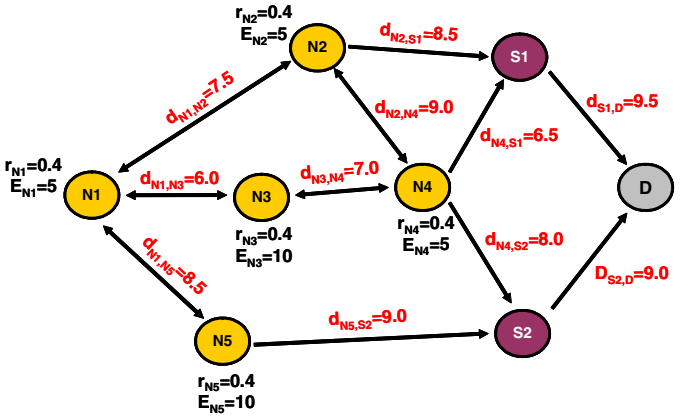


Fig. 3. Network Topology for performance evaluation

Table 3. Flow at each link with power management.

$f_{N1,N2}$	$f_{N3,N1}$	$f_{N1,N5}$	$f_{N2,S1}$	$f_{N3,N4}$
0.31	0.21	0.30	0.61	0.19
$f_{N5,S2}$	$f_{N4,S1}$	$f_{N4,S2}$	$f_{N2,N4}$	$f_{S1,D} + f_{S2,D}$
0.70	0.39	0.30	0	2.0

increased. Therefore it has an effect of increasing information flow when each node has the same energy as the method without power control. The proposed routing mechanism with power management is more efficient than the one without power management when nodes are assumed to consume more transmission energy as distance increases (i.e., $\alpha \approx 4$).

Next we evaluate feasible time for randomly generated sensor networks. Nodes are assumed to be positioned randomly in $20m \times 20m$ space. It is assumed that there is only one sink node and the number of nodes is 10 to 50. The initial node energy is 10 and r_i is 0.3 in the simulation. Fig. 4 compares the feasible time of the proposed method with the method without power management as the number of nodes increases. The feasible time of the proposed method is longer until the number of nodes becomes 30.The reason is that the nodes around the sink node relay the information and the battery drains out at the nodes for the first time. Fig 5 shows the mean residual energy of the nodes as the number of nodes changes. The result presents that the proposed method provides better energy savings than the method without power management. Note that the residual energy of the proposed method is always greater than that of the method without power management although network lifetime becomes the same as the number of nodes becomes more than 30. This indicates that the proposed method can save energy at nodes by power management.

Table 4. Consumed energy with power management during optimal feasible time.

Node	Initial Energy	Energy Consumption	Residual Energy
N1	5	5	0
N2	5	5	0
N3	10	2.22	7.78
N4	5	5	0
N5	10	6.43	3.57

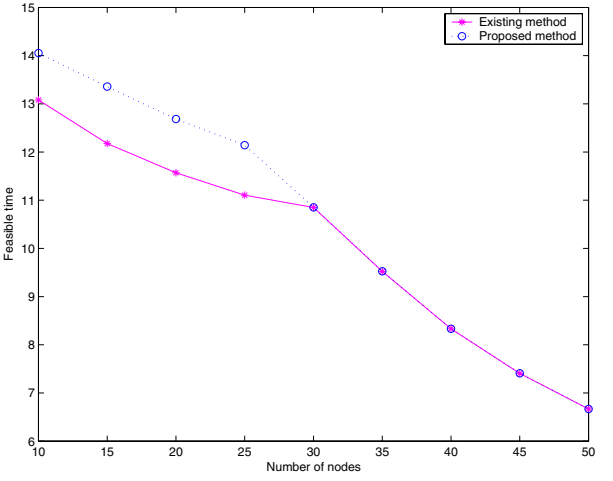


Fig. 4. The feasible time as the number of nodes increases (E_i=10, r_i=0.3).

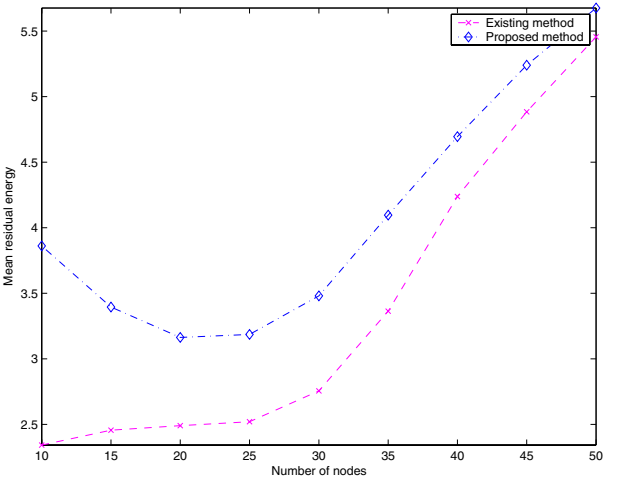


Fig. 5. The mean residual energy as the number of nodes increases (E_i=10, r_i=0.3).

6 Conclusion

We propose an energy efficient routing mechanism by power management to prolong one of the most important factors, network lifetime. In the mechanism, each node is assumed to be able to adjust its transmission power. We model the problem and present an iterative algorithm to find the optimal solution for the maximal network lifetime. The proposed method is shown to have much longer network lifetime than the previous method without power management, and it is very efficient in terms of energy consumption. Network lifetime is determined by the node with little residual energy and/or much information to send or to relay. So in order to further prolong the network lifetime, a mechanism that information flow is not concentrated to some nodes with little energy could be beneficial.

References

1. G. Zussman and A. Segall. Energy Efficient Routing in Ad-Hoc Disaster Recovery Network. In *Proc. of IEEE INFOCOM*, pages 682–691, 2003.
2. J. -H. Chang, and L. Tassiulas. Energy Conserving Routing in Wireless Ad-hoc Network. In *Proc. of IEEE INFOCOM*, vol. 1, pages 26–30, 2000.
3. P. Chen, B. O'Dea, and E. Callaway. Energy Efficient System Design with Optimum Transission Range for Wireless Ad Hoc Networks. In *Proc. of IEEE ICC*, pages 945–952, 2002.
4. A. Cerpa and D. Estrin. ASCENT : Adaptive Self-Configuring Sensor Networks Topologies. In *Proc. of IEEE INFOCOM*, vol. 3, pages 1278–1287, 2002.
5. F. Cheng and L. Zhang. PEAS : A Robust Energy Conserving Protocol for Long-lived Sensor Networks. In *Proc. of Distributed Computing Systems*, pages 28–37, 2003.
6. C. -K. Toh. Maximum Battery Life Routing to Support Ubiquitous Mobile Computing in Wireless Ad Hoc Networks. *IEEE Communications Magazine*, vol. 39, pages 102–114, Jun. 2001.
7. W. Ye, J. Heidemann and D. Estrin. An Energy-Efficient MAC Protocol for Wireless Sensor Networks. In *Proc. of IEEE INFOCOM 2002*, pages 1567–1576, 2002.
8. R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows*. Prentice Hall, 1993.
9. N. Bambos. Toward Power Sensitive Network Architectures in Wireless Communications : Concept, Issues and Design Aspects. *IEEE Personal Communications*, vol. 5, pages 50–59, Jun. 2001.
10. T.-C. Hou. Transmission Range Control in Multihop Packet Radio Networks. *IEEE Trans. on Communications*, vol. 34, pages 38–44, 1986.
11. K. Scott and N. Bambos. Routing and Channel Assignment for Low Power Transmission in PCS. In *Proc. of International Conference on Universal Personal Communications*, vol. 2, pages 498–502, 1996.
12. M. Woo, S. Singh, and C. S. Raghavendra. Power-Aware Routing in Mobile Ad Hoc Networks. In *Proc. of the ACM/IEEE International Conference on Mobile Computing and Networking*, pages 181–190, 1998.

New Parameter for Balancing Two Independent Measures in Routing Path*

Moonseong Kim[1], Young-Cheol Bang[2], and Hyunseung Choo[1]

[1] School of Information and Communication Engineering
Sungkyunkwan University
440-746, Suwon, Korea +82-31-290-7145
{moonseong,choo}@ece.skku.ac.kr
[2] Department of Computer Engineering
Korea Polytechnic University
429-793, Gyeonggi-Do, Korea +82-31-496-8292
ybang@kpu.ac.kr

Abstract. The end-to-end characteristic is an important factor for QoS support. Since network users and their required bandwidths for applications increase, the efficient usage of networks has been intensively investigated for the better utilization of network resources. The distributed adaptive routing is the typical routing algorithm that is used in the current Internet. If the parameter we concern is to measure the delay on that link, then the shortest path algorithm obtains the least delay path P_{LD}. Also, if the parameter is to measure of the link cost, then the shortest path algorithm calculates the least cost path P_{LC}. The delay constrained least cost (DCLC) path problem has been shown to be NP-hard. The path cost of P_{LD} is relatively more expensive than that of P_{LC}, and the path delay of P_{LC} is relatively higher than that of P_{LD} in DCLC problem. In this paper, we propose an effective parameter that is the probabilistic combination of cost and delay. It significantly contributes to identify the low cost and low delay unicasting path, and improves the path cost with the acceptable delay.

1 Introduction

The advanced multimedia technology in company with high speed networks generates a bunch of real-time applications. The significance of real-time transmission has grown rapidly, since high end services such as video conferencing, demand based services (Video, Music, and News on Demand), Internet broadcasting, etc. are popularized. This end-to-end characteristic is an important factor for QoS support. Since network users and their required bandwidths for applications increase, the efficient usage of networks has been intensively investigated for the better utilization of network resources.

* This paper was supported in part by Brain Korea 21 and University ITRC project. Dr. H. Choo is the corresponding author.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 56–65, 2004.
© Springer-Verlag Berlin Heidelberg 2004

The routing is a process computing a path from source to destination and there exist many mechanisms that can satisfy the service requirement when determining a routing path. The distributed adaptive routing is the typical routing algorithm that is used in the current Internet. Unicast routing protocols can be classified into two general types ; distance vector such as routing information protocol (RIP) [3] and link state such as open shortest path first (OSPF) [4]. The distance vector and link state routing protocols are based on Bellman-Ford algorithm [1] and shortest path algorithm such as Dijkstra's [1], respectively. If the parameter we concern is to measure the delay on that link, then the shortest path algorithm obtains the least delay (LD) path denoted by P_{LD}. Meanwhile, if the parameter is a measure of the link cost, then the shortest path algorithm calculates the least cost (LC) path called P_{LC}. But the path cost of P_{LD} is relatively more expensive than that of P_{LC}, and the path delay of P_{LC} is relatively higher than that of P_{LD}. Therefore, there is a trade-off between the P_{LC} and the P_{LD}.

For distributed real-time applications, the path delay should be acceptable and also its cost should be as low as possible. We call it as the delay constrained least cost (DCLC) path problem. It has been shown to be NP-hard [2]. Widyono proposed an optimal centralized delay constrained algorithm, called the constrained Bellman-Ford (CBF) algorithm [9], to solve it. But the CBF is not practical for large networks due to its exponential running time in worst case. Recently Salama proposed a polynomial time algorithm called delay constrained unicast routing (DCUR) [6]. The cost of the path which is computed in [6] is always within 10% from the optimal CBF. At the current node, the DCUR chooses the LD path when the LC path is rejected to prevent the possibility of constructing paths that violate the delay bound. This procedure is simple but if the DCUR frequently takes the next node by the LD path, then the total path cost becomes high cost. As you see, the DCLC is desirable to find a path that considers the cost and the delay together. Even though there is a loss for the cost, two parameters should be carefully negotiated to reduce the delay. This is because the adjustment between the cost and the delay for the balance is important. Hence, we introduce the new parameter that takes in account both the cost and the delay at the same time. Our proposed parameter is superior to others.

The rest of paper is organized as follows. In section 2, we describe the network model and the interval estimation, section 3 presents details of the new parameter. Then we analyze and evaluate the performance of the proposed parameter by simulation in section 4. Section 5 concludes this paper.

2 Preliminaries

2.1 Network Model

We consider a computer network represented by a directed graph $G = (V, E)$, where V is a set of nodes and E is a set of links (arcs). Each link $(i, j) \in E$ is associated with two parameters, namely cost $c_{(i,j)}$ and delay $d_{(i,j)}$. We assume

that cost and delay on each link is asymmetric in general. Given a network G, we define a path as sequence of nodes $u,\ i,\ j,\ \ldots,\ k,\ v$, such that $(u,i),\ (i,j),\ \ldots,$ and (k,v) belong to E. Let $P(u,v) = \{(u,i),\ (i,j),\ \ldots,\ (k,v)\}$ denote the path from node u to node v. If all elements of the path are distinct, then we say that it is a simple path. We define the length of the path $P(u,v)$, denoted by $n(P(u,v))$, as a number of links in $P(u,v)$. Let $\preceq$ be a binary relation on $P(u,v)$ defined by $(a,b) \preceq (c,d) \leftrightarrow n(P(u,b)) \leq n(P(u,d)),\ ^{\forall}(a,b),\ (c,d) \in P(u,v)$. $(P(u,v), \preceq)$ is a totally ordered set. For given a source node $s \in V$ and a destination node $d \in V$, $(2^{s \Rightarrow d}, \infty)$ is the set of all possible paths from s to d.

$$(2^{s \Rightarrow d}, \infty) = \{\ P_k(s,d) \mid \text{all possible paths from } s \text{ to } d,\ ^{\forall}s,\ d \in V,\ ^{\forall}k \in \Lambda\ \}$$

where Λ is a index set.

Both cost and delay of an arbitrary path P_k are assumed to be a function from $(2^{s \Rightarrow d}, \infty)$ to a nonnegative real number $\mathcal{R}^+$. Since $(P_k, \preceq)$ is a totally ordered set, if there exists a bijective function f_k then P_k is isomorphic to $\mathcal{N}_{n(P_k)}$.

$$P_k = \{(u,i),(i,j),\ldots,(k,v)\} \xrightarrow{f_k} \mathcal{N}_{n(P_k)} = \{1,2,\ldots,n(P_k)\}$$

We define

a function of path cost $\phi_C(P_k) = \sum_{r=1}^{n(P_k)} c_{f_k^{-1}(r)}$ and

a function of delay along the path $\phi_D(P_k) = \sum_{r=1}^{n(P_k)} d_{f_k^{-1}(r)}\ ,\ ^{\forall}P_k \in (2^{s \Rightarrow d}, \infty)\ .$

$(2^{s \Rightarrow d}, supD)$ is the set of paths from s to d for which the end-to-end delay is bounded by $supD$. Therefore $(2^{s \Rightarrow d}, supD) \subseteq (2^{s \Rightarrow d}, \infty)$. The DCLC problem is to find the path that satisfies $min\{\ \phi_C(P_k) \mid P_k \in (2^{s \Rightarrow d}, supD),\ ^{\forall}k \in \Lambda\ \}$.

2.2 Statistic Interval Estimation

An interval estimate of a parameter θ is an interval (θ_1, θ_2), the endpoints of which are functions $\theta_1 = g_1(X)$ and $\theta_2 = g_2(X)$ of the observation vector X. The corresponding random interval (θ_1, θ_2) is the interval estimator of θ. We shall say that (θ_1, θ_2) is a γ confidence interval of θ if $Prob\{\theta_1 < \theta < \theta_2\} = \gamma$. The constant γ is the confidence coefficient of the estimate and the difference $\alpha = 1-\gamma$ is the confidence level. Thus γ is a subjective measure of our confidence that the unknown θ is in the interval (θ_1, θ_2) [5]. The $100(1-\alpha)\%$ confidence interval for the sample mean $\bar{X}$ of the X can be described by $(\bar{X} - z_{\alpha/2}\frac{S}{\sqrt{n}}\ ,\ \bar{X} + z_{\alpha/2}\frac{S}{\sqrt{n}})$ when unknown variance and S is sample variance. If we would like to have the 95% confidence interval, then the solution of the following equation is $z_{\alpha/2} = 1.96$ as the percentile which means

$$2\int_0^{z_{\alpha/2}} \frac{1}{\sqrt{2\pi}}\, e^{-\frac{x^2}{2}}\, dx\ =\ 0.95\ .$$

3 Proposed Parameter for Low Cost and Low Delay

3.1 New Parameter

In this paper, we assume that the image of cost function is equal to the image of delay function as a matter of convenience. We compute two paths P_{LD} and P_{LC} from s to d. Since only link-delays are considered to compute $P_{LD}(s,d)$, $\phi_C(P_{LD})$ is always greater than or equal to $\phi_C(P_{LC})$. If the cost of the path, $\phi_C(P_{LD})$, is decreased by $100\big(1-\frac{\phi_C(P_{LC})}{\phi_C(P_{LD})}\big)\%$, $\phi_C(P_{LD})$ is obviously equal to $\phi_C(P_{LC})$. Let $\bar{C}$ be the average of link cost $c_{(i,j)}$ along P_{LD} with $(i,j) \in P_{LD}$ then

$$\bar{C} \;=\; \frac{\phi_C(P_{LD})}{n(P_{LD})} \quad .$$

To decrease $100\big(1-\frac{\phi_C(P_{LC})}{\phi_C(P_{LD})}\big)\%$ for $\phi_C(P_{LD})$, we consider the confidence interval $2\times 100\big(1-\frac{\phi_C(P_{LC})}{\phi_C(P_{LD})}\big)\%$ and should calculate its percentile. Because the normal density function is symmetric to the mean, if the value that has to be decreased is greater than or equal to 50% then we interpret this value as 99.9% confidence interval.

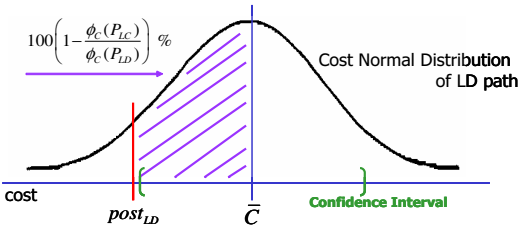


Fig. 1. $post_{LD}$

As shown in Fig. 1, $post_{LD}$ is a datum point to change the link costs at P_{LD}. Thus, it is necessary to find the percentile. In order to obtain it, we can use the cumulative distribution function (CDF). Ideally, the CDF is a discrete function but we assume that the CDF is a continuous function in convenience through out this paper. Let the CDF be $F(x)$ such that

$$F(x) \;=\; \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}}\, e^{-\frac{y^2}{2}}\, dy \quad .$$

Then, the percentile is a solution of the following equation.

$$F(z^d_{\alpha/2}) - \frac{1}{2} \;=\; 1 - \frac{\phi_C(P_{LC})}{\phi_C(P_{LD})}$$

which means

$$z^d_{\alpha/2} \;=\; F^{-1}\big(\frac{3}{2} - \frac{\phi_C(P_{LC})}{\phi_C(P_{LD})}\big) \quad \text{if } 100\big(1 - \frac{\phi_C(P_{LC})}{\phi_C(P_{LD})}\big)\% < 50\% \;.$$

Table 1 shows the percentile we have calculated by the Mathematica.

Table 1. The percentile

$\eta \;=\; [\; 100\;(\;1 - \frac{\phi_C(P_{LC})}{\phi_C(P_{LD})}\;)\;]\;\%$ The function $[x]$ gives the integer closest to x.									
$z_{\alpha/2}$ is $z^c_{\alpha/2}$ or $z^d_{\alpha/2}$. $\quad z_{\alpha/2} = 3.29$ if $\eta \geq 50$									
η	$z_{\alpha/2}$	η	$z_{\alpha/2}$	η	$z_{\alpha/2}$	η	$z_{\alpha/2}$	η	$z_{\alpha/2}$
49	2.33	48	2.05	47	1.88	46	1.75	45	1.65
44	1.56	43	1.48	42	1.41	41	1.34	40	1.28
39	1.23	38	1.18	37	1.13	36	1.08	35	1.04
34	0.99	33	0.95	32	0.92	31	0.88	30	0.84
29	0.81	28	0.77	27	0.74	26	0.71	25	0.67
24	0.64	23	0.61	22	0.58	21	0.55	20	0.52
19	0.50	18	0.47	17	0.44	16	0.41	15	0.39
14	0.36	13	0.33	12	0.31	11	0.28	10	0.25
9	0.23	8	0.20	7	0.18	6	0.15	5	0.13
4	0.1	3	0.08	2	0.05	1	0.03	0	0.00

After calculating the percentile, we compute $post_{LD}$.

$$post_{LD} \;=\; \bar{C} - z^d_{\alpha/2}\frac{S_{LD}}{\sqrt{n(P_{LD})}}$$

where S_{LD} is the sample standard deviation.

$$S_{LD} \;=\; \sqrt{\frac{1}{n(P_{LD})-1}\sum_{r=1}^{n(P_{LD})}(c_{f^{-1}_{LD}(r)} - \bar{C})^2} \;.$$

If $n(P_{LD}) = 1$, then $S_{LD} = 0$. The function f_{LD} is introduced in section 2.1. The new parameter of each link is as follows :

$$Cfct_{(i,j)}(c_{(i,j)}) \;=\; max\{\;1,\; 1 + (c_{(i,j)} - post_{LD})\;\} \;.$$

Meanwhile, P_{LC} is computed by taking into account link-cost only. Because only link-costs are considered to compute $P_{LC}(s,d)$, $\phi_D(P_{LC})$ is always greater than or equal to $\phi_D(P_{LD})$. If $\phi_D(P_{LC})$ is decreased by $100\big(1 - \frac{\phi_D(P_{LD})}{\phi_D(P_{LC})}\big)\%$, then $\phi_D(P_{LC}) = \phi_D(P_{LD})$. Since the new parameter of each link, $(i,j) \in P_{LC}$, can be derived by the same manner used in the case of P_{LD},

$$Dfct_{(i,j)}(d_{(i,j)}) \;=\; max\{\;1,\; 1 + (d_{(i,j)} - post_{LC})\;\} \;.$$

Once the $Cfct_{(i,j)}(c_{(i,j)})$ and the $Dfct_{(i,j)}(d_{(i,j)})$ are found, we compute the value $Cfct_{(i,j)}(c_{(i,j)}) \times Dfct_{(i,j)}(d_{(i,j)})$ for each link of P. The best feasible selection is the link with the lowest cost per delay on initial path P. Briefly, the link with the highest 1/cost per delay could be selected. So then,

$$\frac{\frac{1}{Cfct_{(i,j)}(c_{(i,j)})}}{Dfct_{(i,j)}(d_{(i,j)})} = \frac{1}{Cfct_{(i,j)}(c_{(i,j)}) \times Dfct_{(i,j)}(d_{(i,j)})} .$$

If the value of the above formula is low, the performance should be poor. Thus, links with low value of $Cfct_{(i,j)}(c_{(i,j)}) \times Dfct_{(i,j)}(d_{(i,j)})$ should be selected.

3.2 A Case Study

The following steps explain a process for obtaining new parameter.

Steps to calculate the New Parameter

1. Compute two paths P_{LD} and P_{LC}
2. Compute $\bar{C} = \frac{\phi_C(P_{LD})}{n(P_{LD})}$ and $\bar{D} = \frac{\phi_D(P_{LC})}{n(P_{LC})}$
3. Compute $F^{-1}(\frac{3}{2} - \frac{\phi_C(P_{LC})}{\phi_C(P_{LD})})$ and $F^{-1}(\frac{3}{2} - \frac{\phi_D(P_{LD})}{\phi_D(P_{LC})})$ *i.e.*, $z^d_{\alpha/2}$ and $z^c_{\alpha/2}$
4. Compute $post_{LD} = \bar{C} - z^d_{\alpha/2}\frac{S_{LD}}{\sqrt{n(P_{LD})}}$ and $post_{LC} = \bar{D} - z^c_{\alpha/2}\frac{S_{LC}}{\sqrt{n(P_{LC})}}$
5. Compute $Cfct_{(i,j)}(c_{(i,j)}) = max\{\ 1,\ 1 + (c_{(i,j)} - post_{LD})\ \}$ and $Dfct_{(i,j)}(d_{(i,j)}) = max\{\ 1,\ 1 + (d_{(i,j)} - post_{LC})\ \}$
6. We obtain the new parameter $Cfct_{(i,j)}(c_{(i,j)}) \times Dfct_{(i,j)}(d_{(i,j)})$.

In the following, we illustrate full and detailed instructions on the new parameter with examples ; Fig. 2 (a) shows a given network topology. Link cost and link delay are shown to each link as a pair $(cost, delay)$. To construct a path from source node v_1 to destination node v_6, we consider either link cost or link delay. The paths selected as P_{LC} and P_{LD} are shown in Fig. 2 (b) and (c), respectively. Fig. 2 (d) shows the path computed by the new parameter. We obtain the new parameter as follows.

- $\bar{C} = \frac{8+10}{2} = 9$
- $S_{LD} = \sqrt{\frac{(8-9)^2 + (10-9)^2}{2-1}} = \sqrt{2}$
- $[\ 100\big(1 - \frac{\phi_C(P_{LC})}{\phi_C(P_{LD})}\big)\ ]\% = [100(1 - \frac{10}{18})]\% = [44.44]\% = 44\%.$

  See Table 1. $\therefore\ z^d_{\alpha/2} \approx 1.56$

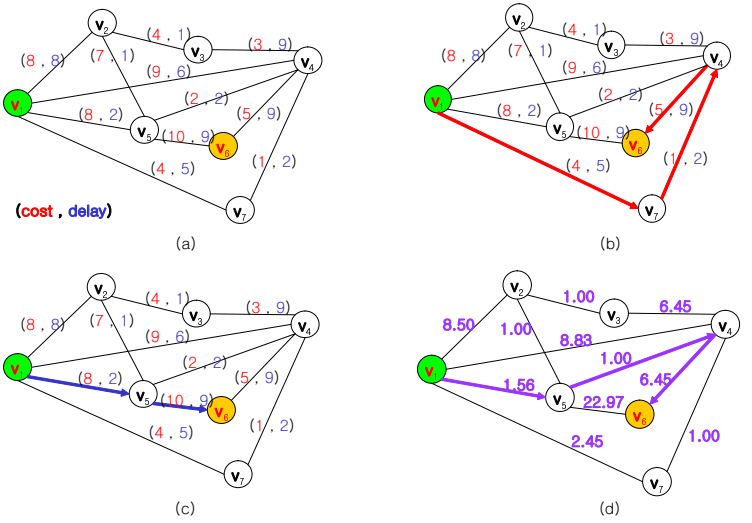


Fig. 2. (a) a given network, (b) least cost path P_{LC}, (c) least delay path P_{LD}, and (d) a path by the new parameter P_{New}

- $post_{LD} = 9 - 1.56 \times \dfrac{\sqrt{2}}{\sqrt{2}} = 7.44$
- $Cfct_{(i,j)}(c_{(i,j)}) = max\{\ 1,\ 1 + (c_{(i,j)} - 7.44)\}$
- $\bar{D} = \dfrac{5+2+9}{3} = 5.33$
- $S_{LC} = \sqrt{\dfrac{(5-5.33)^2 + (2-5.33)^2 + (9-5.33)^2}{3-1}} = \sqrt{12.33}$
- $[\ 100\big(1 - \dfrac{\phi_D(P_{LD})}{\phi_D(P_{LC})}\big)\ ]\% = [100(1 - \dfrac{11}{16})]\% = [31.25]\% = 31\%.$

  See Table 1. $\therefore\ z^c_{\alpha/2} \approx 0.88$
- $post_{LC} = 5.33 - 0.88 \times \dfrac{\sqrt{12.33}}{\sqrt{3}} = 3.55$
- $Dfct_{(i,j)}(d_{(i,j)}) = max\{\ 1,\ 1 + (d_{(i,j)} - 3.55)\ \}$
- $Cfct_{(i,j)}(c_{(i,j)}) \times Dfct_{(i,j)}(d_{(i,j)})$

In Fig. 2 (d), we calculate $Cfct_{(v_1,v_5)} = max\{\ 1,\ 1 + (8 - 7.44)\} = 1.56$ and $Dfct_{(v_1,v_5)} = max\{\ 1,\ 1 + (2 - 3.55)\} = 1$ at link (v_1, v_5). By the same manner, we obtain all new parameters in the network. Fig. 2 (d) shows the path constructed by the new parameter. $P_{New}(v_1, v_6) = \{(v_1, v_5), (v_5, v_4), (v_4, v_6)\}$.

We know that $\phi_C(P_{LC}) \leq \boldsymbol{\phi_C(P_{New})} \leq \phi_C(P_{LD})$ and $\phi_D(P_{LD}) \leq \boldsymbol{\phi_D(P_{New})} \leq \phi_D(P_{LC})$ in Table 2. Namely, although $\phi_C(P_{LC})$ is the smallest of all, $\phi_D(P_{LC})$ is $100 \cdot (16-11)/11 = 45.5\%$ worse than $\phi_D(P_{LD})$. Also $\phi_D(P_{LD})$

Table 2. The comparison with example results

P_{LC}		P_{LD}		P_{New}	
$\phi_C(P_{LC})$	$\phi_D(P_{LC})$	$\phi_C(P_{LD})$	$\phi_D(P_{LD})$	$\phi_C(P_{New})$	$\phi_D(P_{New})$
10	**16**	**18**	11	***15***	***13***

is the lowest of all, but $\phi_C(P_{LD})$ is $100 \cdot (18-10)/10 = 80\%$ worse than $\phi_C(P_{LC})$. If we use the new parameter that adjusts balance between the cost and the delay at the same time, then its path cost is $100 \cdot (18-15)/18 = 16.7\%$ lower than $\phi_C(P_{LD})$ and its path delay is $100 \cdot (16-13)/16 = 18.8\%$ lower than $\phi_D(P_{LC})$.

4 Performance Evaluation

We compare our new parameter to only link-delays and only link-costs as you see in Table 2. Two performance measures - $\phi_C(P)$ and $\phi_D(P)$ - are combined our concern and investigated here.

First we describe the generation of random network topologies for the evaluation and the simulation results based on the network topology generated. The details of the generation for random network topologies are as follows. The method uses parameters n - the number of nodes in networks, and P_e - the probability of link existence between any node pair [7,8]. Let us remark that if a random graph models a random network then this graph should be connected. Hence, the graph should contain at least a spanning tree. So, firstly a random spanning tree is generated. As we know, we consider cases for $n \geq 3$. A tree with 3 nodes is unique, and thus we use this as an initial tree. And we expand to a spanning tree with n nodes. After adjusting the probability P_e, we generate other non-tree links at random for the graph based network topology.

Let us calculate the adjusted probability P_e^a. By $Prob\{event\}$ denote a probability of the event. Suppose e is a possible link between a couple of nodes, then we have

$$\begin{aligned} P_e &= Prob\{\ e \in \textit{spanning tree}\ \} + Prob\{\ e \notin \textit{spanning tree}\ \} \cdot P_e^a \\ P_e &= \frac{n-1}{n(n-1)/2} + (1 - \frac{n-1}{n(n-1)/2}) \cdot P_e^a \\ \therefore\ P_e^a &= \frac{nP_e - 2}{n-2}\ . \end{aligned}$$

Let us describe a pseudo code for random network topologies. Here A is an incident matrix, r is a simple variable, and $random()$ is a function producing uniformly distributed random values between 0 and 1.

Graph Generation Algorithm

Begin
$A_{1,2} = A_{2,1} = A_{2,3} = A_{3,2} = 1$
For $i = 4$ to n **Do**
  $r = (i-1) \times random() + 1$
  $A_{r,i} = A_{i,r} = 1$

For $i = 1$ to $(n-1)$ **Do**
  For $j = (i+1)$ to n **Do**
    If $P_e > random()$ **Then** $A_{i,j} = A_{j,i} = 1$
End Algorithm.

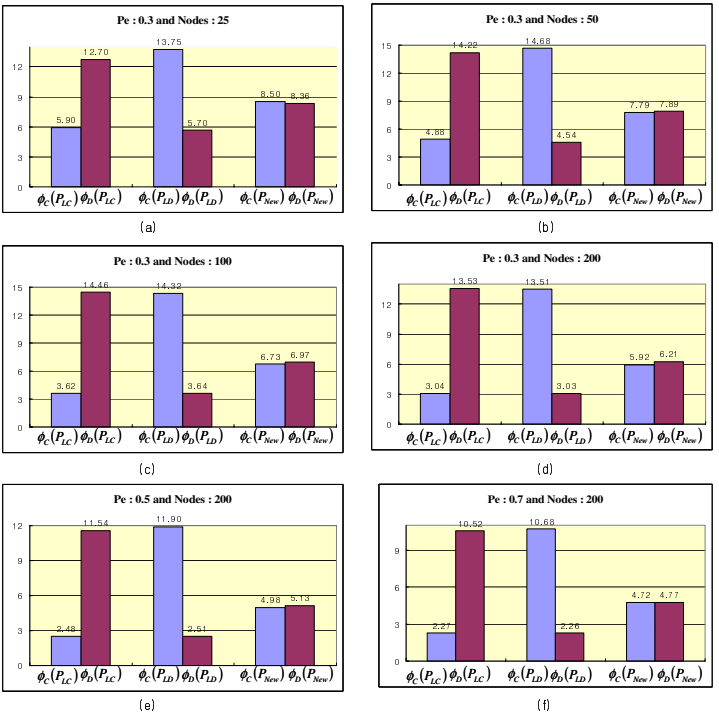


Fig. 3. Performance comparison for each P_e and n

We now describe some numerical results with which we compare the performance for the new parameter. The proposed one is implemented in C++. We consider networks with number of nodes which is equal to 25, 50, 100, and 200. We generate 10 different networks for each size given above. The random networks used in our experiments are directed, symmetric, and connected, where each node in networks has the probability of links (P_e) equal to 0.3, 0.5, and 0.7. Randomly selected source and destination nodes are picked uniformly. Each costs and delays are uniformly random integer values between 0 and 10. We simulate 1000 times ($10 \times 100 = 1000$) for each n and P_e.

Fig. 3 shows the average $\phi_C(P)$ and $\phi_D(P)$, where each path P is P_{LC}, P_{LD}, and P_{New}. As a result, the proposed new parameter ascertains that $\phi_C(P_{LC}) \leq \boldsymbol{\phi_C(P_{New})} \leq \phi_C(P_{LD})$ and $\phi_D(P_{LD}) \leq \boldsymbol{\phi_D(P_{New})} \leq \phi_D(P_{LC})$. For details on analyzing performance for the new parameter, refer to Fig. 3 (d). The path cost $\phi_C(P_{LC}) = 3.04$ is far superior, and $\phi_C(P_{LD}) = 13.51$ is the worst. Likewise the path delay $\phi_D(P_{LD}) = 3.03$ is far better, and $\phi_D(P_{LC}) = 13.53$ is

the highest. Let us consider path P_{New} which is measured by the probabilistic combination of cost and delay at the same time. Because the $\phi_C(P_{New})$ occupies $\frac{5.92 - 3.04}{13.51 - 3.04} \times 100 = 27.5\%$ between $\phi_C(P_{LC})$ and $\phi_C(P_{LD})$, $\phi_C(P_{New})$ is somewhat expensive than $\phi_C(P_{LC})$ but becomes more superior than $\phi_C(P_{LD})$. In the same manner, the $\phi_D(P_{New})$ occupies $\frac{6.21 - 3.03}{13.53 - 3.03} \times 100 = 30.3\%$ between $\phi_D(P_{LD})$ and $\phi_D(P_{LC})$. In other words, the new parameter takes into account both cost and delay at the same time. It significantly contributes to identify the low cost and low delay unicasting path and the performance improvement.

5 Conclusion

The distributed adaptive routing is very important in the current Internet. If an application requires a certain QoS based on the delay, the shortest path algorithm calculates the least delay path. In the meantime, if the application pursues cost effective data transmission, the least cost path should be calculated. In this paper, we have formulated the new parameter for the DCLC path problem, which is known to be NP-hard [2]. Because the DCLC must consider together cost and delay at the same time, P_{LC} and P_{LD} are unsuitable to the DCLC problem. Hence the new parameter takes into consideration both cost and delay at the same time. We would like to extend the new parameter to the weighted parameter that can regulate as wanted $\phi_C(P)$ and $\phi_D(P)$. In addition, we present unicasting and multicasting algorithm for the DCLC path problem by using the proposed new parameter in the future.

References

1. D. Bertsekas and R. Gallager, Data Networks, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1992.
2. M. Garey and D. Johnson, Computers and intractability: A Guide to the Theory of NP-Completeness, New York: Freeman, 1979.
3. C. Hedrick, "Routing information protocol," http://www.ietf.org/rfc/rfc1058.txt, June 1988.
4. J. Moy, "OSPF Version 2," http://www.ietf.org/rfc/rfc1583.txt, March 1994.
5. A. Papoulis and S. U. Pillai, Probability, Random Variables, and Stochastic Processes, 4th ed. McGraw-Hill, 2002.
6. D.S. Reeves and H.F. Salama, "A distributed algorithm for delay-constrained unicast routing," IEEE/ACM Transactions on Networking, vol. 8, pp. 239-250, April 2000.
7. A.S. Rodionov and H. Choo, "On Generating Random Network Structures: Trees," Springer-Verlag Lecture Notes in Computer Science, vol. 2658, pp. 879-887, June 2003.
8. A.S. Rodionov and H. Choo, "On Generating Random Network Structures: Connected Graphs," International Conference on Information Networking 2004, Proc. ICOIN-18, pp. 1145-1152, February 2004.
9. R. Widyono, "The Design and Evaluation of Routing Algorithms for Real-Time Channels," International Computer Science Institute, Univ. of California at Berkeley, Tech. Rep. ICSI TR-94-024, June 1994.

A Study on Efficient Key Distribution and Renewal in Broadcast Encryption*

Deok-Gyu Lee and Im-Yeong Lee

Division of Information Technology Engineering, Soonchunhyang University,
#646 Eupnae-ri, Shinchang-myun, Asan-si, Choongchungnam-do, 336-745, Korea
{hbrhcdbr, imylee}@sch.ac.kr
http://sec-cse.sch.ac.kr

Abstract. The method of broadcast encryption has been applied to the transmission of digital information such as multimedia, software, and paid TV on the open network. In this broadcast encryption method, only previously authorized users can gain access to digital information. When broadcast message is transmitted, authorized users can first decode the session key using the previously given private key and get digital information using this session key. This way, users retrieve a message or a session key using the key transmitted by broadcasters. For their part, broadcasters need to generate and distribute keys. Broadcasters should also carry out efficient key renewal when users subscribe or unsubscribe. This paper introduces how to generate and distribute key efficiently and how key renewal works. The proposal uses two methods: (1) the server generates keys without the consent of users by anticipating users, and; (2) the server and users generate keys by mutual agreement. The advantage of the two proposed methods is that the receiver can decode broadcast message using a secret key. Even if the key is renewed later, the user can efficiently renew using only a single set of information.

1 Introduction

The broadcast encryption method has been recently applied to the transmission of digital information such as multimedia, software, pay TV, etc. As one of the key providing methods, the public key method uses a single group key to encode the session key and an infinite number of keys for decoding. As such, the server encodes the session key and enables each user to decode it using different keys. In the broadcast encryption method, only previously authorized users can gain access to digital information. When broadcast message is transmitted, authorized users can first decode the session key using the previously given private key and get digital information using this session key. In short, broadcast encryption involves generating, distributing, and renewing keys.

* This work was supported by grant No. R05-2003-000-12019-0 from the Basic Research Program of Korea Science & Engineering Foundation

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 66–76, 2004.
© Springer-Verlag Berlin Heidelberg 2004

This paper introduces the method of generating, distributing, and renewing keys efficiently. The proposal uses 2 methods: (1) the server generates keys without the consent of users by anticipating users, and; (2) the server and users generate keys by mutual agreement. The advantage of the two proposed schemes is that the receiver can decode broadcast message using a secret key. Even if the key is renewed later, the user can efficiently renew using only a single set of information. In the proposed methods, key renewal factor is added for fast key renewal. This allows easy key renewal and provides users with renewal values even in case of new subscription or withdrawal. This paper briefly introduces application methods in broadcast encryption, goes through the existing methods, and discusses each stage of the proposed methods. Likewise, the protocols of each stage are explained. Proposed methods are also reviewed through comparison analysis between the existing methods and the proposed methods. Finally, the conclusion is presented.

2 Overview of Broadcast Encryption

2.1 Application Methods

Broadcast encryption is based on two models. Although there are some differences between the applied models, each of them will be discussed. To begin with, the first model is shown in the figure below:

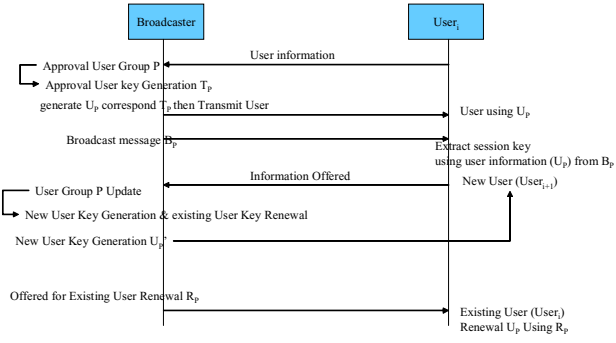


Fig. 1. Application Method 1

This method involves generating/distributing keys using information between the user and server. This is similar to the existing multicast method, since the message provided is determined by the previous user group. The only difference lies in the transmitting method. The user participation time may be included in the key generating time, since it requires user participation in the process of key generation. Unlike the abovementioned method, the server in the second applied model generates keys.

The server generates keys by anticipating user participation at its own discretion. This method enables quick creation and renewal since the server generates all users' keys without their consent. In case the server becomes the target of attacks or other vicious purposes, however, it becomes very vulnerable.

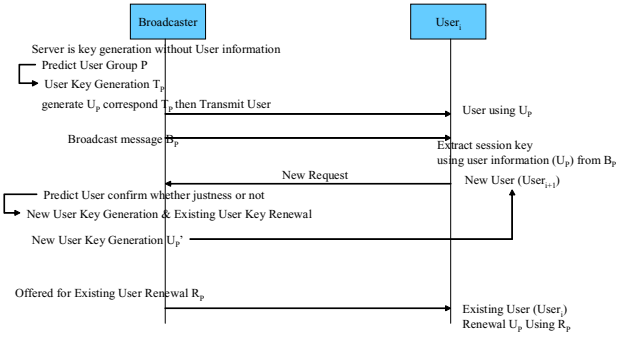


Fig. 2. Application Method 2

3 Conventional Scheme – Narayanan

The Narayanan method suggests a practical paid TV scheme based on RSA, which has the ability to trace vicious users. The method of tracing vicious users can be carried out using the following principle:

When composing n *number of* $(t+1)$ *vectors* $X_1, X_2, \ldots, X_n$ *with linear combination of arbitrary number of* $s(<t)$ *vectors, there is a high probability of finding the correct vectors used.*

3.1 Protocol of the Narayanan Scheme

Assume one contents provider broadcasting in m number of channels and n number of users. Protocol is divided into seven algorithms such as Setup, AddStream, AddUser, Broadcast, Receive, Subscribe, and Unsubscribe. Whether or not users receive channels can be displayed with Subsc and a $m \times n$ matrix. If user U_j is registered at S_i, the value of $Subsc[i, j]$ is 1. Otherwise, if the user is not registered, the value is 0.

Algorithm Setup
The contents provider generates the following variables:

When $N = pq, R, d_r \le R\{1,2,\ldots,\varphi(N)\}$, $1 \le r \le 4+t$. P and q are larger prime numbers, and R is a random value. p, q, and d are composed as secret keys of the contents provider. In turn, the contents provider opens the public key (N).

Algorithm AddStream

The contents provider randomly choose $g_i \in Z_{N^*}$ to add new channel stream S_i to the system and sets up $Subsc[i, j]$ to set all j to 0; thus preventing the opening of the g_i value.

Algorithm AddUser

The contents provider chooses $(e_{1j}, e_{2j}, \ldots, e_{(t+4)j})$, which satisfies $\sum_{r=1}^{t+4} e_{rj} d_r = R\Phi(N) + 1$. At this time, U_j receives the decoding device (Set-Top Terminal) that stored the secret key in the safe memory. The secret key of U_j will be $(e_{1j}, e_{2j}, \ldots, e_{(t+4)j})$.

Algorithm Subscribe

When user U_j subscribes to service S_i, the contents provider transmits g_i^{e1j} to U_j and changes the $Subsc\ [i, j]$ value to 1.

Algorithm Unsubscribe

When user U_j unsubscribe to S_i, the contents provider sets $Subsc\ [i, j] = 0$. Similar to the AddStream algorithm, the contents provider chooses a new g_i value and transmits g_i^{e1j} to all users who have the value $Subsc\ [i, j] = 1$.

Algorithm Broadcast

To transmit message M to channel stream S_i, the contents provider randomly chooses value x as a value smaller than $\Phi(N)$ and transmits encrypted data $C = (x, C_1, C_2, \ldots, C_{t+4})$ as $C_1 = M^{d1} g_i^x, C_2 = M^{d2}, C_{t+4} = M^{dt+4}$.

Algorithm Receive

User U_j determines $\left(\prod_{r=1}^{t+4} C_r^{e_{rj}}\right) / g_i^{xe_{1j}}$ using secret key $(e_{1j}, e_{2j}, \ldots, e_{(t+4)j})$ to decode encrypted data $C = (x, C_1, C_2, \ldots, C_{t+4})$, which is transmitted to channel stream S_i. User U_j restores contents data M by going through this process.

$$\left(\prod_{r=1}^{t+4} C_r^{e_{rj}}\right) / g_i^{xe_{1j}} = M^{R\Phi(N)+1} = M$$

Problems of the Narayanan Scheme

The Narayanan scheme requires the traffic of $(x, C_1, C_2, \ldots, C_{t+4})$ per channel. Since traffic is related to the number of channels, increasing number of channels can also cause heavier traffic. In addition, despite managing to find traitor U_j, the contents provider has to distribute a new secret key to all subscribers again except U_j to disqualify U_j.

4 Proposed Method

Methods for efficient key renewal are proposed in a situation wherein existing users unsubscribe and new users subscribe. The proposal is largely divided into two methods: (1) the server generates and distributes keys for encrypted communication, anticipating users without their consent, and; (2) the server generates the encrypted broadcasting key only upon obtaining users' consent.

4.1 Overview of Proposed Methods

This section presents an overview of the proposed methods. Figure 3 is a classification of scenarios that can occur using the proposed methods. The scenario is composed of the basic flow, renewal flow, new process flow, leaving flow, and flow of false user anticipation. The proposal can be classified into three large parts depending on the scenario: key generation and distribution, broadcast message generation, and key renewal. Similarly, two proposed methods can be applied to the entire flow. Differences are only found in the initial key generation and distribution part through server anticipation and users; the rest proceeds in the same manner.

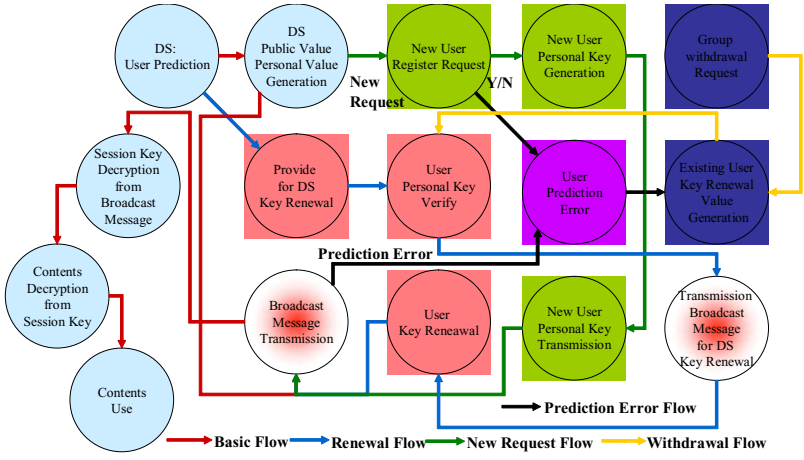


Fig. 3. Proposed Scheme Whole Flows

In addition, the first method in the proposal has the following features: (1) the user's private key is generated by the server; (2) persons other than the user cannot decode the broadcasting message, and; (3) renewing keys is easy, which is important when new subscribers subscribe and existing users unsubscribe. On the other hand, in the second method, the user's private key is generated only upon obtaining the user's consent. When many users gather, the server generates a public key. Through the public key, the encrypted broadcasting message is transmitted. Likewise, subscribing

and unsubscribing can take place easily by deleting the information provided by the user.

4.2 System Coefficient

The following is a description of the system coefficient used in this method:

p : Prime number(≥ 512 bit) q : Prime number(≥ 160 bit $(q \mid p-1)$)

l : Number for Personal Key Generation o : Security parameter

$d_1, \ldots, d_k$: List of Personal Decryption Key

M : Message S : Session Key

k : User e : Public Encryption Key

r_i : Set of Random Number ($r_i \in Z_p$): $(r_1, \ldots, r_k)$, $h_i = g^{r_i}$

$\langle y, h_1, \ldots, h_k \rangle$: Public Key: $y = \prod_{i=1}^{k} h_i^{a_i}$ $B = M \cdot (orS)\, y^{aT}$, $H_i = \prod_{i=1}^{k} h_1^{a}$

$d_i = \theta_i \cdot \gamma^{(i)}$ $\left(\gamma^{(i)} \in \Gamma\right)$: $\Gamma = \gamma_1, , \gamma_k$ a : Random Element($a \in Z_q$)

C : Broadcast message: $C = < M \cdot (orS)\, y^{aT}, h_1^a, \ldots h_k^a > = < B, H_1, \ldots, H_k >$

a_i : Random Number ($a_i \in Z_q$) $(a_1, \ldots, a_k)$

T : Element for Key Renewal ($t_1, \ldots, t_k \in Z_q$), $T = t_1 \cdot \ldots \cdot t_k$

b : user's generated public information($b \in Z_p$)

ζ : User is random choose value Ξ : Stored User of ID

4.3 Protocol-1

1) Key generation and distribution stage

Key generation is processed by the server. The generation and transmission of the private and public keys will go through the following process:

Step 1. The server anticipates users and randomly chooses string accordingly.

$$i = 1, \ldots, k \text{ prediction} \rightarrow r_i \text{ row choose} \tag{1}$$

Step 2. Based on this chosen string, the server generates the values required to produce the public key.

$$h_i = g^{r_i} \bmod q \text{ Compute, Public Key } \langle y, h_1, \ldots, h_k \rangle \tag{2}$$

T Generated For renewal: $T = t_1 \cdot \ldots \cdot t_k$

Step 3. The server produces the public key using the created value h and calculates the private key.

$$\theta_i = \left(\sum_{j=1}^{k} r_j a_j t_j \right) / \left(\sum_{j=1}^{k} r_j \gamma_j \right) \bmod q \tag{3}$$

Step 4. The server transmits the generated private key d_i to user.

$$d_i = \theta_i \cdot \gamma_i \tag{4}$$

Step 5. The user acquires θ_i from the received d_i.

$$d_i = \theta_i \cdot \gamma_i / \gamma_i \tag{5}$$

2) Broadcast message generation stage

Broadcast messages can be transmitted by encrypting the session key with the encrypted message and encrypting the message itself. Both methods are described as follows:

Step 1. The server calculates by encrypting message M or session key S.

Step 2. The server randomly chooses factor a, operates key renewal factor T, and uses both random factor and renewal factor to produce a message.

Step 3. The server produces and transmits the broadcast message.

$$C =< M(S)y^{aT}, h_1^a, , h_k^a > \tag{6}$$

Step 4. The received message acquires message M or session key S using the private key.

$$M(S) = C / U^{\theta_i}, U = \prod_{j=1}^{k} H_j^{\gamma_j} \tag{7}$$

$$U^{\theta_i} = \left(\prod_{j=1}^{k} H_j^{\gamma_j}\right) = \left(\sum_{j=1}^{k} g^{ar_j\gamma_j}\right)^{\theta_j} = \left(\sum_{j=1}^{k} g^{r_j\gamma_j}\right)^{a\cdot\theta_j} = \left(\sum_{j=1}^{k} g^{d_j\gamma_j}\right)^{a} = \left(\sum_{j=1}^{k} h_j^{d_j T}\right)^{a} = y^{aT}$$

$$M(S) = M(S) \cdot y^{aT} / y^{aT}$$

3) Key renewal stage

In case of existing users who unsubscribe or new users who subscribe, the following process is carried out:

Step 1. User i requests for withdrawal.

Step 2. The server removes i's renewal factor from renewal factor T to update existing users' private keys.

Step 3. After removal, the server renews private keys and re-transmits them to users.

$$\theta_i \cdot \gamma^{(i)} \cdot t_i^{-1} = d_i' \tag{8}$$

Step 4. Users get broadcast message using the renewed keys and acquire message by decoding the encrypted message as follows:

$$M(S) = B / U^{\theta_i t_i^{-1}}, U = \prod_{j=1}^{k} H_j^{\gamma_j} \tag{9}$$

Using $\left(C = \langle B, H_1, \ldots, H_K \rangle\right) = \left(C = \left\langle M(orS) \cdot y^{aTt_i^{-1}}, h_1^a, ..., h_k^a \right\rangle\right)^{\theta_i}$ compute

$$U^{\theta_i t_i^{-1}} = \left(\prod_{j=1}^{k} H_j^{\gamma_j}\right)^{\theta_i t_i^{-1}} = \left(\sum_{j=1}^{k} g^{ar_j\gamma_j}\right)^{\theta_i t_i^{-1}} = \left(\sum_{j=1}^{k} g^{r_j\gamma_j}\right)^{a\theta_i t_i^{-1}} = \left(\sum_{j=1}^{k} g^{r_j d_j t_i}\right)^{at_i^{-1}} = \left(\prod_{j=1}^{k} H_j^{d_j t_i}\right)^{a} = y^{aTt_i^{-1}}$$

$$M(S) = M(S) \cdot y^{aTt_i^{-1}} / y^{aTt_i^{-1}}$$

4.4 Protocol-2

1) Key generation and distribution stage

Key generation is processed by the server. The generation and transmission of private and public keys will go through the following process:

Step 1. The data provider generates value to acquire and open user information.

$$\gamma_{(i)} = (\gamma_1, \ldots, \gamma_k) \in \Gamma \tag{1}$$

Step 2. User calculates the following value using opened Γ and his or her own ID :

$$ID_i \equiv (\Xi_i)^{\gamma_i} (\text{mod } n) \tag{2}$$

Step 3. The following value is calculated using the produced value:

$$\Xi_i \equiv (ID_i)^{1/\gamma_i} (\text{mod } n),\ U \equiv \Xi_i \cdot \zeta (\text{mod } n),\ \Theta \equiv \zeta^b (\text{mod } n) \tag{3}$$

Step 4. The server transmits the values (Θ, U) produced by the user to the data provider.

Step 5. The data acquires user information ID_i using the values (Θ, U) provided.

$$\begin{gathered} \text{Extract } \zeta \text{ from } \Theta \equiv \zeta^b (\text{mod } n) \\ \text{Compute } \Xi_i \text{ from } U \equiv \Xi i \cdot \zeta (\text{mod } n) \text{ by } \zeta \\ \text{Compute } \Xi_i \equiv (ID_i)^{1/\gamma_i} (\text{mod } n), \text{ and acquire } ID_i \equiv (\Xi_i)^{\gamma_i} (\text{mod } n) \end{gathered} \tag{4}$$

Step 6. The server chooses the string of ID_i using the formation of user i and calculates the following:

$$\begin{gathered} h_i = g^{r_i} \bmod q \quad \text{Compute, Public Key } \langle y, h_1, \ldots, h_k \rangle \\ T \text{ Generated For renewal: } T = t_1 \cdot \ldots \cdot t_k \end{gathered} \tag{5}$$

Step 7. The server generates the public key using the created value h and calculates the private key accordingly. Equal to equation (3)

Step 8. The server transmits the generated private key d_i to the user. Equal to equation (4)

Step 9. User acquires θ_i from the transmitted d_i . Equal to equation (5)

2) Broadcast message generation stage

Broadcast messages can be transmitted by encrypting the session key with the encrypted message and encrypting the message itself. Both methods are described as follows:

Step 1. The server calculates by encrypting message M or session key S .

Step 2. The server randomly chooses factor a, operates key renewal factor T, and uses both random factor and renewal factor to produce a message.
Step 3. The server produces and transmits the broadcast message. Equal to equation (6)
Step 4. The received message acquires message M or session key S using the private key. Equal to equation (7)

3) Key renewal stage
In case of existing users who unsubscribe or new users who subscribe, the following process is carried out:
Step 1. User i requests for withdrawal.
Step 2. The server removes i's renewal factor from the renewal factor T to update the existing users' private keys.
Step 3. After removal, the server renews private keys and re-transmits them to users. Equal to equation (8)
Step 4. Users get broadcast message using the renewed keys and acquire message by decoding the encrypted message as follows: Equal to equation (9)

5 Comparison Analysis between the Conventional Scheme and Proposed Scheme

This paper proposes the broadcast encryption method, which is more efficient than the existing method in generating and renewing keys. The stability of the proposed method is based on discrete algebra issue. Compared to the existing method, the proposed method achieves efficiency in user participation, key renewal, user withdrawal, or operating amount. In this section, the efficiency of the proposed method is presented vis-à-vis the existing method.

User Participation
In the existing method, the server anticipates users, generates keys in advance without user participation, and provides and distributes them to new users who subscribe. In this method, when an attack is made on the server itself, all keys created by the server can be affected.

Key renewal
In the existing Key Pre-distribution Scheme (KPS), message is transmitted as encrypted using this scheme after the key is generated and distributed. When the session is closed after the user checks the transmitted message, a key is newly produced and transmitted. If an attack is made on the key, all keys will be re-generated instead of merely renewing them. In the proposed method, however, keys are ready to use after renewing the existing users' keys in case of subscription or withdrawal.

Table 1. Comparison Analysis Between the Conventional Scheme and Proposed Scheme

	User Partici-pation	Key Renewal	No. N with-drawal	Continuity of key	Traitor Tracing	Re-operation due to false prediction error
Convention KPS[2]	X	X	X	O	X	O
Broadcast Encryption[1]	X	O	X	□	X	O
IKPS[4]	O	O	X	□	X	X
Proposed Scheme – I	X	O	O	O	X	X
Proposed Scheme - II	O	O	O	O	X	X

Re-operation due to false prediction error

In the existing method and the proposed method - I, the server should set up and control the system. If the server controls flexible users, the anticipation of users should be carried out correctly. Therefore, the server should implement re-operation or additional operation in case initial anticipation fails. In the existing method, however, there is no such operation in case of failure of user anticipation. In the proposed method, user anticipation can be achieved smoothly through a simple operation like g^r when the server configures the system. Likewise, random number r can be generated on Z_p. Problems can also be solved by giving numbers larger than the expected number of users in advance.

6 Conclusion

Broadcast encryption is used to provide contents only for authorized users on the open network. Except authorized users, nobody can obtain messages from the broadcast message; authorized users can obtain the session key, with the private key transmitted in advance. This paper proposes the method of generation, distribution, and renewal of private key and suggests an easier way of renewing after users' requests for withdrawal or process of the server's withdrawal for existing users. Further studies on user tracing and key cycling are recommended.

References

1. Amos Fiat, and Moni Naor, "Broadcast Encryption", Crypto'93, LNCS 773, 480-491
2. C. Blundo, Luiz A. Frota Mattos, D.R. Stinson, "Generalized Beimel-Chor schemes for Broadcast Enryption and Interactive Key Distribution", Crypto'96, LNCS 1109
3. Carlo Blundo, Luiz A. Frota Mattos, and Douglas R. Stinson, " Trade-offs Between Communication and Storage in Unconditionally Secure Schemes for Broadcast Encryption and Interactive Key Distribution", Crypto 98

4. Juan A. Garay, Jessica Staddon, and Avishai Wool, "Long-Lived Broadcast Encryption", Crypto'00, LNCS 1880, 333-352
5. Ignacio Gracia, Sebastia Martin, and Carles Padro, "Improving the Trade-off Between Storage and Communication in Broadcast Encryption Schemes", 2001
6. Dani Halevy, and Adi Shamir, "The LSD Broadcast Encryption Scheme," Crypto'02, LNCS 2442, 47-60
7. Yevgeniy Dodis and Nelly Fazio, "Public Key Broadcast Encryption for Stateless Receivers", DRM2002, 2002. 11. 18
8. Donald Beaver, and Nicol So, "Global, Unpredictable Bit Generation Without Broadcast," 1993
9. Michel Abdalla, Yucal Shavitt, And Avishai Wool, "Towards Marking Broadcast Encryption Practical", FC'99, LNCS 1648
10. Dong Hun Lee, Hyun Jung Kim, and Jong In Lim, "Efficient Public-Key Traitor Tracing in Provably Secure Broadcast Encryption with Unlimited Revocation
11. A. Narayanan, "Practical Pay TV schemes," to appear in the Proceedings of ACISP03, July, 2003

Self-Tuning Mechanism for Genetic Algorithms Parameters, an Application to Data-Object Allocation in the Web*

Joaquín Pérez[1], Rodolfo A. Pazos[1], Juan Frausto[2], Guillermo Rodríguez[3], Laura Cruz[4], Graciela Mora[4], and Héctor Fraire[4]

[1] Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET)
AP 5-164, Cuernavaca, Mor. 62490, México
{jperez, pazos}@sd-cenidet.com.mx
[2] ITESM, Campus Cuernavaca, México
AP C-99 Cuernavaca, Mor. 62589, México
juan.frausto@itesm.mx
[3] Instituto de Investigaciones Eléctricas, IIE
gro@iie.org.mx
[4] Instituto Tecnológico de Ciudad Madero, México
{hfraire,lcruzreyes}@prodigy.net.mx

Abstract. In this paper, a new mechanism for automatically obtaining some control parameter values for Genetic Algorithms is presented, which is independent of problem domain and size. This approach differs from the traditional methods which require knowing first the problem domain, and then knowing how to select the parameter values for solving specific problem instances. The proposed method is based on a sample of problem instances, whose solution permits to characterize the problem and to obtain the parameter values.To test the method, a combinatorial optimization model for data-objects allocation in the Web (known as DFAR) was solved using Genetic Algorithms. We show how the proposed mechanism permits to develop a set of mathematical expressions that relates the problem instance size to the control parameters of the algorithm. The experimental results show that the self-tuning of control parameter values of the Genetic Algorithm for a given instance is possible, and that this mechanism yields satisfactory results in quality and execution time. We consider that the proposed method principles can be extended for the self-tuning of control parameters for other heuristic algorithms.

1 Introduction

A large number of real problems are NP-complete combinatorial optimization problems. These problems require the use of heuristic methods for solving large size instances of the problems. Genetic Algorithms (GA) constitute an alternative that has been used for solving this kind of problems [1].

* This research was supported in part by CONACYT and COSNET.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 77–86, 2004.
© Springer-Verlag Berlin Heidelberg 2004

A framework used frequently for the study of evolutionary algorithms includes: the population, the selection operator, the reproduction operators, and the generation overlap. The GA's components have control parameters associated. The choice of appropriate parameters setting is one of the most important factors that affect the algorithms efficiency. Nevertheless, it is a difficult task to devise an effective control parameter mechanism that obtains an adequate balance between quality and processing time. It requires a profound knowledge of the nature of the problem to be solved, which is not usually trivial.

For several years we have been working on the distribution design problem and the design of solution algorithms. We have carried out a large number of experiments with different solution algorithms, and a recurrent problem is the tuning of the algorithm control parameters; hence our interest in incorporating self-tuning mechanisms for parameter adjustment. In [2]. we proposed an on-line method to set the control parameters of the Threshold Accepting algorithm. However, with that method we can not relate algorithm parameters to the problem size. Now, we want to explore, with genetic algorithms, the off-line automatic configuration of parameters.

2 Related Work

Diverse works try to establish the relationship between the values of the genetic algorithm control parameters and the algorithm performance.

The following are some of the most important investigation works on the application of the theoretical results in practical methodologies.

Back uses an evolutionary on-line strategy to adjust the parameter values [3]. Mercer and Grefenstette use a genetic meta-algorithm to evolve the control parameter values of another genetic algorithm [4,5]. Smith uses an equation derived from the theoretical model proposed by Goldberg [6]. Harik uses a technique prospection based [7], for tuning the population size using an on-line process.

Table 1 summarizes research works on parameter adaptation. It shows the work reference, applied technique and on-line controlled parameters (population size P, crossover rate C and mutation rate M).

Table 1. Parameter adaptation work summary

Ref.	Tech.	P	C	M
[3]	Evolution		√	√
[4]	Meta-algorithm		√	√
[5]	Meta-algorithm		√	√
[6]	Theoretical model	√		
[7]	Prospection	√		

We propose a new method to obtain relationships between the problem size and the population size, generation number, and the mutation rate. The process consists of applying off-line statistical techniques to determine mathematical expressions for the relationships between the problem size and the parameter values. With this approach it is possible to tune a genetic algorithm to solve many problem instances at a cost lower than that of the prospection approach.

3 Proposed Method for Self-Tuning GA Parameters

In this work we propose the use of off-line sampling to get the relationship between the problem size and the control parameters of a Genetic Algorithm. The self-tuning mechanism is constructed iteratively by solving a set of problem instances and gathering statistics of algorithm performance to obtain the relationship sought. With this approach it is possible to tune a genetic algorithm for solving many problem instances at low cost. To automate the configuration of the algorithm control parameters the following procedure was applied:

Iteratively carries out next steps:

Step 1. Record instances. Keep a record of all the instances currently solved with the GA configured manually. For each instance, its size, configuration used, and the corresponding performance are recorded.

Step 2. Select a representative sample. Get a representative sample of recorded instances, each one of different size. The sample is built considering only the best configuration for each selected instance.

Step 3. Determine correlation functions. Get the relationship between the problem size and the algorithm parameters

Step 4. Feedback. The established relationships reflect the behavior of the recorded instances. When new instances with a different structure occur, the adjustment mechanism can lose effectiveness.

The proposed method allows advancing toward an optimal parameter configuration with an iterative and systematic approach. An important advantage of this method is that the experimental costs are reduced gradually. We can start using an initial solved instance set and continue adding new solved instances. In the next section we describe an application problem to explain some method details.

4 Application Problem

To test the method, a combinatorial optimization model for data-objects allocation in the Web (known as DFAR) was solved using Genetic Algorithms. We show how the proposed method permits to develop a set of mathematical expressions that relates the problem instance size to the control parameters of the algorithm. In this section we describe the distribution design problem and the DFAR mathematical model.

4.1 Problem Description

Traditionally it has been considered that the distributed database (DDB) distribution design consists of two sequential phases. Contrary to this widespread belief, it has been shown that it is simpler to solve the problem using our approach which combines both phases [8]. A key element of this approach is the formulation of a mathematical model that integrates both phases. In order to describe the model and its properties, the following definition is introduced: $DB-object$: Entity of a database that requires to be allocated, which can be an attribute, a relation or a file. They are independent units that must be allocated in the sites of a network.

The DDB distribution design problem consists of allocating DB-objects, such that the total cost of data transmission for processing all the applications is minimized. New allocation schemas should be generated that adapt to changes in usage and access patterns of read applications, which prevent the system degradation. A formal definition of the problem is given below.

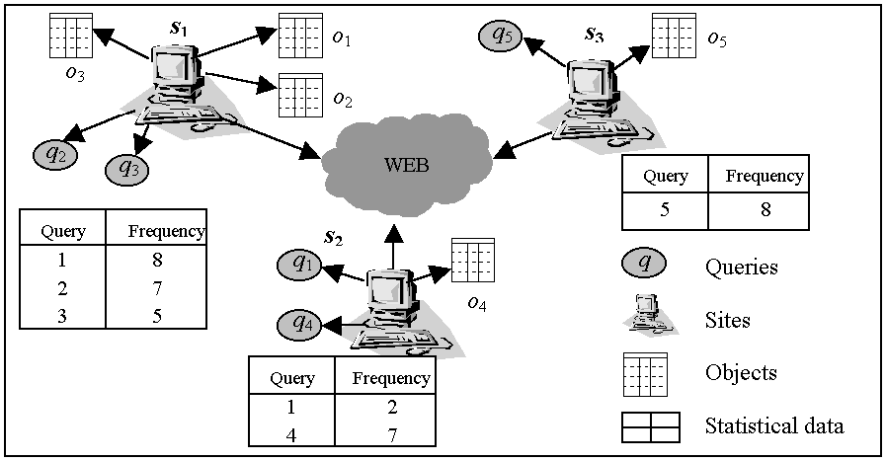


Fig. 1. Distribution Design Problem

Assume there are a set of DB-objects $O = \{o_1, o_2, ..., o_n\}$, a computer communication network that consists of a set of sites $S = \{s_1, s_2, ..., s_n\}$, where a set of queries $Q = \{q_1, q_2, ..., q_n\}$ are executed, the DB-objects required by each query, an initial DB-object allocation schema, and the access frequencies of each query from each site in a time period. The problem consists of obtaining a new allocation schema that adapts to a new database usage pattern and minimizes transmission costs. Figure 1 shows the main elements related with this problem.

4.2 Objective Function

The integer (binary) programming model consists of an objective function and four intrinsic constraints. In this model the decision about storing a DB-object m in site j is represented by a binary variable x_{mj}. Thus, $x_{mj} = 1$ if m is stored in j, and $x_{mj} = 0$ otherwise.

The objective function below (1) models costs using four terms: 1) the transmission cost incurred for processing all the queries, 2) the cost for accessing multiple remote DB-objects required for query processing, 3) the cost for DB-object storage in network sites, and 4) the transmission cost for migrating DB-objects between nodes.

$$\min z = \sum_k \sum_i f_{ki} \sum_m \sum_j q_{km} l_{km} c_{ij} x_{mj} + \sum_i \sum_k \sum_j c1 f_{ki} y_{kj}$$

$$+ \sum_j c2 w_j + \sum_m \sum_i \sum_j a_{mi} c_{ij} d_m x_{mj}. \quad (1)$$

where

f_{ki}	=	emission frequency of query k from site i, during a given period of time;
q_{km}	=	usage parameter, $q_{km} = 1$ if query k uses DB-object m, otherwise $q_{km} = 0$;
l_{km}	=	number of packets for transporting DB-object m for query k;
c_{ij}	=	communication cost between sites i and j;
$c1$	=	cost for accessing several remote DB-objects for processing a query;
y_{kj}	=	indicates if query k accesses one or more DB-objects located at site j;
$c2$	=	cost for allocating DB-objects in a site;
w_j	=	indicates if there exist DB-objects at site j;
a_{mi}	=	indicates if DB-object m was previously located in site i;
d_m	=	number of packets for moving DB-object m to another site.

4.3 Intrinsic Constraints of the Problem

The model solutions are subject to four constraints: each DB-object must be stored in one site only, each DB-object must be stored in a site that executes at least one query that uses it, a constraint to determinate for each query where is the DB-objects required, and a constraint to determinate if the sites contains DB-objects. The detailed formulation of the constraints can be found in [2,8].

5 Implementation

In this section we present some application examples of the proposed method, using the DDB design problem.

5.1 Record Instances

Table 2 shows four entries of the historical record. These correspond to an instance solved using a manually configured GA. Columns 1 and 2 contain the instance identifier I and the instance size S in bytes. Columns 3-6 show the configuration of four GA parameters (population size P, generation number G, crossover rate C, and mutation rate M). Columns 7 and 8, present the algorithm performance (the best solution B found by the GA, and the execution time T in seconds).

Table 2. Parameter adaptation work summary

I	S	P	G	C	M	B	T
P8	921620	30	300	1	0	415899.7	2.81
P8	921620	30	300	0.9	0.1	408754.9	2.61
P8	921620	30	300	0.9	0.01	385483.3	2.62
P8	921620	375	19200	0.9	0.01	61188.0	128.52

Table 2 shows the best solutions that were obtaining with the specified configurations.

5.2 Select a Representative Sample

If the number of solved instances is not very large, all the available instances can be included in the sample; otherwise, it is necessary to use some sampling technique. Table 3 presents an example of a sample of instances of different size extracted from the record, where column headings have the same meaning as those of Table 2. For each selected instance only its best configuration is included in the sample.

Table 3. Instances representative sample

I	S	P	G	M	B	T
P1	108	30	300	0	302.2	0.015
P2	308	30	300	0	604.4	0.017
P3	1044	30	300	0	1208.8	0.021
P4	3860	30	300	0	2417.6	0.032
P5	14868	30	300	1	4835.2	0.085
P6	58388	60	1200	1	9670.4	0.467
P7	231444	150	4800	1	19340.8	8.137
P8	921620	375	19200	1	61188.0	128.5
P9	3678228	750	96000	1	185679.0	1543.5

5.3 Determine Correlation Functions

Population Correlation Functions. To find the relationship between the problem size and the population size we used two techniques: statistical regression and estimate based on proportions. Three mathematical expressions (2,3,4) were constructed to determinate the population P size in function of the problema size x. The expressions contain derived coefficients of the lineal and logarithmic statistical estimates and a constant of proportionality.

$$\textit{Linear estimate}: \qquad P(x) = 0.00019843x + 56.7506298. \tag{2}$$

$$\textit{Logarithmic estimate}: \qquad P(x) = 45.7182388\ (1.00000087)^{x}. \tag{3}$$

$$\textit{Proportional estimate}: \qquad P(x) = 2^{(\log_4 x)\ 0.938523962}. \tag{4}$$

At this point we considered that the exponential estimate had a shape that fitted best the real data graph and that a fine adjustment of the function parameters could improve the quality of the estimation. Finally the exponential relationship was adjusted to get the best estimation. As a result of the fine adjustment the following adjustment factors were defined: $\alpha = 14868, \beta = 309$.

$$p(x) = \begin{cases} 30 & if \quad x \leq \alpha \\ \frac{x}{\lfloor \beta * 2^{\lceil \log_4 (x/\alpha) \rceil} \rfloor} & if \quad x > \alpha. \end{cases} \tag{5}$$

Figure 2 shows the graphs of the real data and the adjusted proportional estimate.

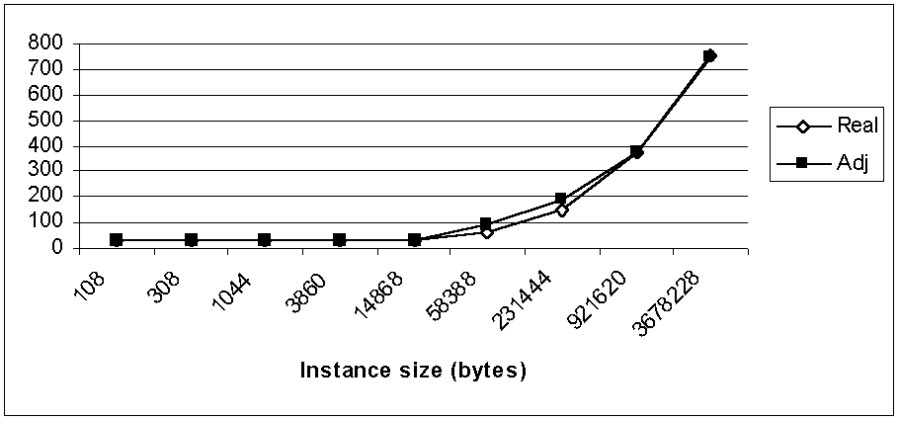


Fig. 2. Correlation functions graphs

Correlation Functions for the Generation Number and Mutation Rate. Similarly the relationships between the size of the problem, and the number of generations and the mutation rate were determined. Expressions 6 and 7 specify the relationship between the instance size and these algorithm parameters. In these expressions, G is the number of generations, and M is the mutation rate and $\delta = 4.8$, is an adjust parameter.

$$G(x) = \begin{cases} 300 & if \quad x \leq \alpha \\ \frac{[P(x)]^2}{\delta} & if \quad x > \alpha. \end{cases} \tag{6}$$

$$M(x) = \begin{cases} 0 & if \quad x \leq \alpha \\ 1 & if \quad x > \alpha. \end{cases} \tag{7}$$

As can be observed, the parameter tuning mechanism is defined using an off-line procedure. The evaluation and subsequent use of this mechanism should be carried out on-line. In this example, for the evaluation of the mechanism a comparative experiment was carried out using a GA configured manually according to the recommendations proposed in the literature. In Figure 3 the comparative results of the solution quality for the instances sample can be observed. The execution time, to solve all instances, was similar for both algorithms. The algorithm configured with the designed mathematical expressions was able to obtain a better solution than the algorithm configured according to the literature.

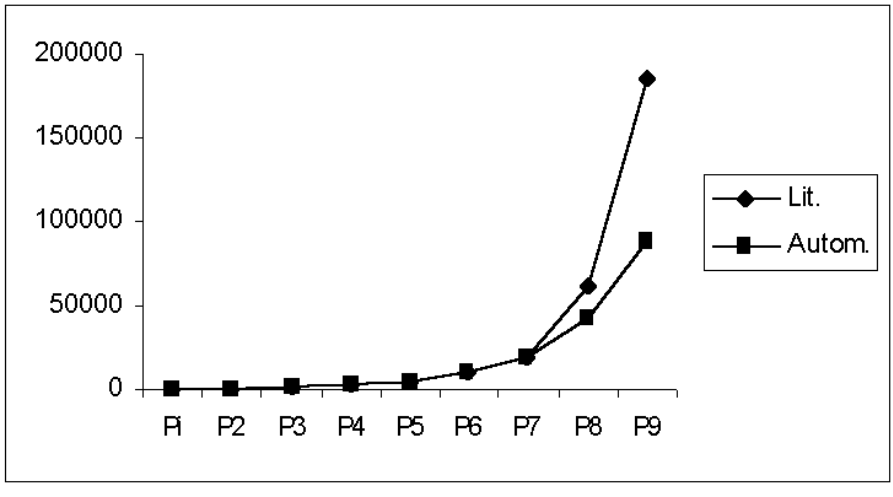


Fig. 3. Quality solution tests

5.4 Feedback

Since the tuning mechanism requires a periodic refinement, the performance of the GA configured automatically can be compared versus other algorithms when solving new instances. If for some instance another algorithm is superior, the GA will be configured manually to equal or surpass the performance of the other algorithm. The instance and their different configurations are recorded in the historical record and the tuning process is repeated from step 2 through step 4. Hence the experimental cost it is relatively low, because it takes advantage of all the experimental results stored in the historical record.

6 Conclusions and Future Work

In this work, we propose a new method to obtain relationships between the problem size and the population size, generation number, and the mutation rate. The process consists of applying off-line statistical techniques to determine mathematical expressions for these relationships. The mathematical expressions are used on-line to control the values of the algorithm parameters. With this approach it is possible to tune a genetic algorithm to solve many problem instances at a cost lower than other approaches.

We present a genetic algorithm configured with mathematical expressions, designed with the proposed method, which it was able to obtain a better solution than the algorithm configured according to the literature.

Currently the self-tuning GA is being tested for solving a new model of the DDB design problem that incorporates data replication, and the preliminary results are encouraging.

Up to now we have adjusted independently the parameters that depend on the characteristics of the instances and parameters that depend on the size of the problem. In the near future we are planning to devise a self-tuning mechanism for adjusting simultaneously both types of parameters.

References

1. Fogel, D., Ghozeil, A.: Using Fitness Distributions to Design More Efficient Evolutionary Computations. Proceedings of the 1996 IEEE Conference on Evolutionary Computation, Nagoya, Japan. IEEE Press, Piscataway N.J. (1996) 11-19
2. Pérez, J., Pazos, R.A., Velez, L. Rodriguez, G.: Automatic Generation of Control Parameters for the Threshold Accepting Algorithm, Lectures Notes in Computer Science, Vol. 2313. Springer-Verlag, Berlin Heidelberg New York (2002) 119-127.
3. Back, T., Schwefel, H.P.: Evolution Strategies I: Variants and their computational implementation. In: Winter, G., Périaux, J, Galán, M., Cuesta, P. (eds.): Genetic Algorithms in Engineering and Computer Science. Chichester: John Wiley and Sons. (1995) Chapter 6, 111-126
4. Mercer, R.E., Sampson, J.R.: Adaptive Search Using a Reproductive Meta-plan. Kybernets 7 (1978) 215-228

5. Grefenstette, J.J.: Optimization of Control Parameters for Genetic Algorithms. In: Sage, A.P. (ed.): IEEE Transactions on Systems, Man and Cybernetics, Volume SMC-16(1). New York: IEEE (1986) 122-128
6. Smith, R.E., Smuda, E.: Adaptively Resizing Population: Algorithm Analysis and First Results. Complex Systems 9 (1995) 47-72
7. Harik, G.R., Lobo, F.G.: A parameter-less Genetic Algorithm. In: Banzhaf, W., Daida, J., Eiben, A.E., Garzon, M.H., Honavar, V., Jakiela. M., Smith, R.E. (eds.): Proceedings of the Genetic and Evolutionary Computation Conference GECCO-99. San Francisco, CA: Morgan Kaufmann (1999) 258-267
8. Pérez, J., Pazos, R.A., Romero, D., Santaolaya, R., Rodrí guez, G., Sosa, V.: Adaptive and Scalable Allocation of Data-Objects in the Web. Lectures Notes in Computer Science, Vol. 2667. Springer-Verlag, Berlin Heidelberg New York (2003) 134-143

Digit-Serial AB^2 Systolic Array for Division in GF(2^m)*

Nam-Yeun Kim and Kee-Young Yoo

Department of Computer Engineering, Kyungpook National University,
Daegu, Korea 702-701
knyeun@hanmail.net yook@knu.ac.kr

Abstract. Digit-serial architecture is an attractive solution for systems requiring moderate sample rate and where area and time consumption are critical. The current paper presents a digit-serial-in-serial-out systolic architecture for performing an AB^2 operation in GF(2^m). If the appropriate digit-size is selected, the proposed method can meet the throughput requirement of a specific application with minimum hardware. And, the area-time complexity of the pipelined digit-serial AB^2 systolic architecture is approximately 10.9% lower than that of the nonpipelined version when m = 160 and L = 2. Based on the new AB^2 digit-serial architecture, we also proposed a digit-serial systolic for inverse/divisions. Furthermore, since the proposed architectures are simplicity, regularity, modularity and pipelinability, they are well suited to VLSI, and can also be utilized as the basic architecture for a cryptoprocessor.

1 Introduction

Arithmetic in finite fields GF(2^m) are widely used in public-key cryptography [1, 2]. The key arithmetic operations involved in cryptography are multiplication, power-sum (AB^2+C), inverse/division, and exponentiation. Among these operations, a power-sum is known as an efficient basic operation for public-key cryptosystems [3]. For example, the division is performed using multiplication and a multiplicative inverse, that is $A/B = AB^{-1}$, while the inverse can be regarded as a special case of exponentiation, because $B^{-1} = B^{2^m-2} = (B(B(B\cdots B(B(B)^2)^2\cdots)^2)^2)^2$, where AB^2 operation can be used to compute. However, since an inverse operation is quite time consuming, a high-speed circuit is preferable for such operations.

For a digit-serial system, the data words are first partitioned into digits of some bits each and then processed and transmitted on a digit-by-digit basis [4]. Suppose the word size is m-bits, the digit size is L-bits, and $N = \lceil m/L \rceil$, then bit-parallel and bit-serial systems process the input data at a rate of m-bits and 1-bit per clock cycle, respectively, while a digit-serial system processes the input data at a rate of L-bits per clock cycle. Therefore, if the appropriate digit size is chosen, a digit-serial architecture can meet the throughput requirement of a certain application with minimum hardware.

In this paper, we proposed the digit-serial-in-serial-out systolic implementation of an AB^2 and A/B architecture in GF(2^m) using the standard basis. The latency and area

* This research was supported by University IT Research Center Project.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 87–96, 2004.
© Springer-Verlag Berlin Heidelberg 2004

complexity of the proposed architecture is shorter than conventional architectures, plus it is well suited to VLSI implementation and can be easily applied to inverse architecture.

2 Algorithm

A finite field GF(2^m) has 2^m elements and, in this paper, all the (2^m-1) non-zero elements of GF(2^m) are represented using the standard basis. Let $A(x)$ and $B(x)$ be two elements in GF(2^m), that is $A(x) = a_{m-1}x^{m-1} + a_{m-2}x^{m-2} \cdots + a_1x + a_0$, and $B(x) = b_{m-1}x^{m-1} + b_{m-2}x^{m-2} \cdots + b_1x + b_0$, where a_i and $b_i \in$ GF(2) ($0 \le i \le m$-1). A finite field of GF(2^m) elements is generated by a primitive polynomial of degree m over GF(2). Let $F(x)$ be an irreducible polynomial that generates the field and is expressed as $F(x) = x^m + f_{m-1}x^{m-1} + \cdots + f_1x + f_0$, where $f_i \in$ GF(2) ($0 \le i \le m$). Each element in GF(2^m) is a residue mod $F(x)$ and all coefficients are obtained by taking the results modulo 2.

With the fact that

$$B^2(x) = b_{m-1}x^{2(m-1)} + b_{m-2}x^{2(m-2)} + \cdots + b_1x^2 + b_0 = B(x^2) \tag{1}$$

define

$$R(x) = A(x)B^2(x) \bmod F(x) = r_{m-1}x^{m-1} + r_{m-2}x^{m-2} + \cdots + r_1x + r_0 \tag{2}$$

To compute an $[A(x)B^2(x) \bmod F(x)]$ operation, the proposed algorithm starts with computing following recursive equation:

$$\begin{aligned} R(x) &= A(x)B^2(x) \bmod F(x) \\ &= A(x)(b_{m-1}x^{2m-2} + b_{m-2}x^{2m-4} + \cdots + b_1x^2 + b_0) \bmod F(x) \\ &= (A(x)b_{m-1}x^{2m-2} + A(x)b_{m-2}x^{2m-4} + \cdots + A(x)b_1x^2 + A(x)b_0) \bmod F(x) \\ &= (\cdots(\cdots((A(x)b_{m-1})x^2 \bmod F(x) + A(x)b_{m-2})x^2 \bmod F(x) + \cdots \\ &\quad + A(x)b_{m-i})x^2 \bmod F(x) + \cdots + A(x)b_1)x^2 \bmod F(x) + A(x)b_0 \end{aligned} \tag{3}$$

In the recursive form of equation 3, Wang et al. [5] derived the following algorithm:

[Algorithm 1] Wang's Algorithm for $A(x)B^2(x) \bmod F(x)$ [5]
Input: $A(x)$, $B(x)$ and $F(x)$
Output: $R(x) = A(x)B^2(x) \bmod F(x)$
$P_0(x) = 0$
1: for $i = 1$ to m
2: $P_i(x) = P_{i-1}(x) \cdot x^2 \bmod F(x) + A(x)b_{m-i}$
3: $R(x) = P_m(x)$

Wang's algorithm calculates the $[A(x)B^2(x) \bmod F(x)]$ operation by computing a normal AB^2 multiplication part and a modular reduction part all together. Thereby, this algorithm has regularity, while it has to compute a needless modular reduction in the first term, resulting in high area and time cost in hardware implementation. In

addition, when it is tried to implement bit-serial and digit-serial architectures based on Wang's algorithm, it is impossible to derive serial architectures and to expand the digit-size to a $L \times L$ size of the regular square form, instead of 1×2 size, due to the problem of data dependency. To improve the disadvantage of Wang's algorithm, we propose the following AB^2 algorithm.

[Algorithm 2] Proposed algorithm for $A(x)B^2(x) \bmod F(x)$
Input : $A(x)$, $B(x)$ and $F(x)$
Output : $R(x) = A(x)B^2(x) \bmod F(x)$
$P_0(x) = 0$
1: for $i = 1$ to m-1
2: $D_i(x) = (P_{i-1}(x) + A(x)b_{m-i})\, x^2$
3: $P_i(x) = D_i(x) \bmod F(x)$
4: $R(x) = D_m(x) = P_{m-1}(x) + A(x)b_0$

In the proposed Algorithm 2, we use the fact that there is a needless modular reduction of the first term in Wang's algorithm, where modulo reduction is time-consuming calculation. Here, we compute x^2 before the current step by one step. As such, the last term is very simple as it only computes the Ab_0 operation, which does not require any modulo reduction operation. Therefore, it is possible to reduce the area and time complexity in hardware implementation. As shown in the Algorithm 2, we compute two part operations separately, the normal AB^2 operation part and the modular reduction part to obtain the $[A(x)B^2(x) \bmod F(x)]$ result.

With bit-level operations, Algorithm 2 can be rewritten, where the intermediate results $D_i(x)$ and $P_i(x)$ are polynomials of degree at most m+1 and m-1 with coefficients over GF(2), respectively. Here, we define

$$D_i(x) = d^i_{m-1}x^{m+1} + d^i_{m-2}x^m + \cdots + d^i_1 x^3 + d^i_0 x^2 \text{ for } 1 \le i \le m\text{-}1 \tag{4}$$

$$D_m(x) = d^m_{m-1}x^{m-1} + d^m_{m-2}x^{m-2} + \cdots + d^m_1 x^1 + d^m_0 \text{ for } i = m \tag{5}$$

$$P_i(x) = p^i_{m-1}x^{m-1} + p^i_{m-2}x^{m-2} + \cdots + p^i_1 x^1 + p^i_0 \text{ for } 1 \le i \le m \tag{6}$$

In the general terms (i = 1 to m-1), let

$$D_i(x) \quad = (P_{i-1}(x) + A(x)b_{m-i})x^2 = [\sum_{k=0}^{m-1} (p^{i-1}_k + a_k b_{m-i})x^k x^2] \tag{7}$$

From equation 4 and equation 7, we have $d^i_k = p^{i-1}_k + a_k b_{m-i}$, where $p^0_k = 0$, for k = m-1 downto 0.

Define

$$x^m \bmod F(x) \equiv f_{m-1}x^{m-1} + f_{m-2}x^{m-2} + \cdots + f_1 x + f_0 \tag{8}$$

$$x^{m+1} \bmod F(x) = x^m\, x \bmod F(x) \tag{9}$$
$$\equiv f'_{m-1}x^{m-1} + f'_{m-2}x^{m-2} + \cdots + f'_1 x + f'_0 \text{ where, } f'_i \in \text{GF(2)}$$

Substituting equation 8 and 9 into equation 7, the modular reduction operation can be performed as follows:

$$\begin{aligned} P_i(x) &= (d^i_{m-1}x^{m+1} + d^i_{m-2}x^m + d^i_{m-3}x^{m-1} + \cdots + d^i_1 x^3 + d^i_0 x^2) \bmod F(x) \\ &= (d^i_{m-1}f'_{m-1} + d^i_{m-2}f_{m-1} + d^i_{m-3})x^{m-1} + (d^i_{m-1}f'_{m-2} + d^i_{m-2}f_{m-2} \\ &\quad + d^i_{m-4})x^{m-2} + \cdots + (d^i_{m-1}f'_1 + d^i_{m-2}f_1)x + (d^i_{m-1}f'_0 + d^i_{m-2}f_0) \\ &= \sum_{k=0}^{m-1} p^i_k x^k \end{aligned} \tag{10}$$

In the general terms (i=1 to m-1), from equation 7 to equation 10, we can obtain

$$\begin{cases} d^i_k = p^{i-1}_k + a_k b_{m-i} \\ p^i_k = d^i_{m-1}f'_k + d^i_{m-2}f_k + d^i_{k-2} \end{cases} \quad \text{for } i = 1, 2, \cdots, m\text{-}1,\ k = m\text{-}1, m\text{-}2, \cdots, 0 \tag{11}$$

where $p^0_k = 0$, for $k = m$-1, m-2, ···,0 and $d^i_{-1} = 0$, $d^i_{-2} = 0$, for $i = 1, 2, \cdots, m$-1.

Finally ($i = m$), let

$$D_m(x) = P_{m-1}(x) + A(x)b_0 = \sum_{k=0}^{m-1} p^{m-1}_k x^k + \sum_{k=0}^{m-1} a_k b_0 x^k = \sum_{k=0}^{m-1} (p^{m-1}_k + a_k b_0)x^k \tag{12}$$

By comparing equation 5 and equation 12, we can derive

$$d^m_k = p^{m-1}_k + a_k b_0 \ , \text{ for } k = m\text{-}1, m\text{-}2, \cdots, 0 \tag{13}$$

Thus the product $R(x)$ for $[A(x)B^2(x) \bmod F(x)]$ in GF(2^m) can be efficiently computed using the above equations 11 and 13.

3 Systolic AB^2 Architecture

The AB^2 algorithms proposed in the previous section can be illustrated by a two-dimensional systolic power(AB^2) multiplier, denoted by SPM, as shown in Figure 1, where one delay element (denoted by "•") is placed at each horizontal path.

The SPM consists of m^2 cells, which includes $m(m\text{-}1)$ PE1(Processing Element 1) cells and m PE2 cells for AB^2 in GF(2^m), as shown in Figure 2, that are governed by the previous equations 11 and 13, respectively.

Note that the bottom cell circuit in SPM is very simple and reduces the total cell complexity compared to previous architectures. Since the vertical path of each cell only requires three delay elements, except for the cells in the bottom row, the latency is slightly less than the $4m$ units proposed by Wei [6]. However, due to the original characteristics of the (mod $F(x)$) operation in the proposed AB^2 algorithm, there is a two clock cycle delay between the computation of the same order coefficient in two adjacent iterations, which is denoted by two-clock-cycle-gap problem. Furthermore,

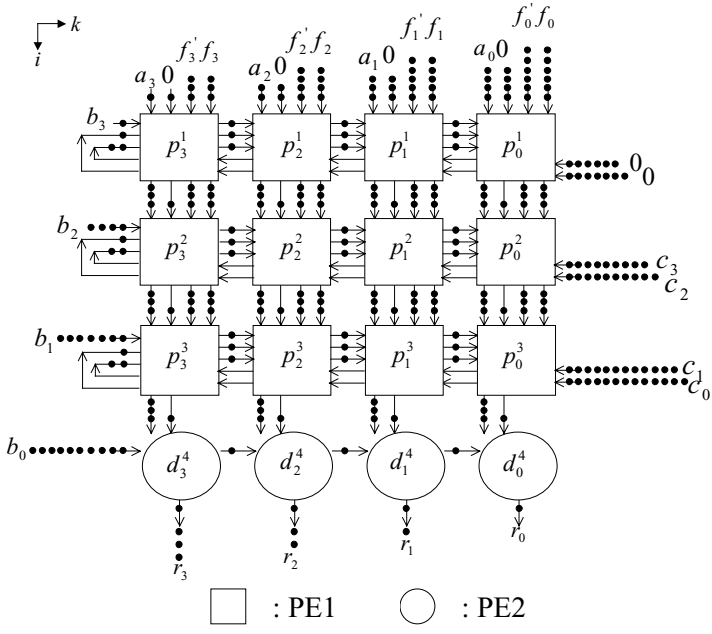


Fig. 1. SPM in GF(2^4).

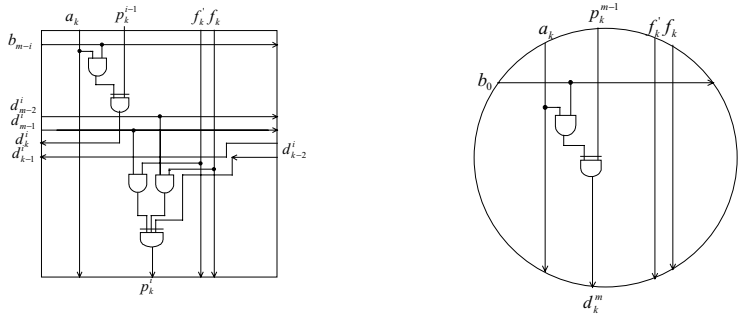


Fig. 2. Circuits of PE1 and PE2 in Figure 1.

the SPM has a bi-directional data flow in a horizontal direction. As described in [7], a system with a unidirectional data flow has several advantages over a system with a bi-directional data flow in terms of the chip cascadability, fault tolerance, and possible wafer-scale integration.

To overcome these problems, an alternative AB^2 multiplication architecture is proposed based on partitioning and merging the previous SPM architecture.

First, partitioning is applied to the SPM architecture. With the exception of the bottom cells, all cells are partitioned into two cells to calculate d_k^i and p_k^i, where the upper layer cells compute d_k^i, while the lower layer cells compute p_k^i. Second, merging is proposed based on the partitioned SPM, as denoted by MSPM. To avoid

the two-clock-cycle-gap problem, Wang and Guo [8] merged two adjacent basic cells in the horizontal direction, producing $m \times m/2$ digit cells. Although this solves the two-clock-cycle-gap problem, the data dependency means that it is impossible to expand the digit cell size from a 1×2 size to the regular square form $L \times L$ size. Therefore, to further improve the performance of the architecture and avoid this problem, the cells in the partitioned SPM are merged in a specific way, where d_k^i and d_{k-1}^i are grouped together, d_{k-2}^i, d_{k-3}^i, p_k^i, and p_{k-1}^i are grouped together, and finally, p_1^i and p_0^i are grouped together, as denoted by PE_A, PE_B, and PE_C, respectively.

Then, the merged architecture is reshaped by applying a coordinate transformation to the index space without changing the cell function. In the cell computing d_k^i, the cell index (i, k) is moved to position $(i, -2i+k+2)$, while in the cell computing p_k^i, the cell index (i, k) is moved to position $(i, -2i+k)$. The resulting DG is shown in Figure 3, when L=2 and m=4., where PE_A, PE_B, and PE_C are represented by the circular, rectangular, and triangular dashed-line, respectively.

In the MSPM, cell merging is used to pre-calculate some of the operations, thereby removing the idle cycles in the partitioned SPM. This removal of the idle cycles thus increases the computation efficiency when dealing with dependent multiplications. And, it can be seen that the MSPM involves a unidirectional data flow in the horizontal directional, instead of a bi-directional data flow.

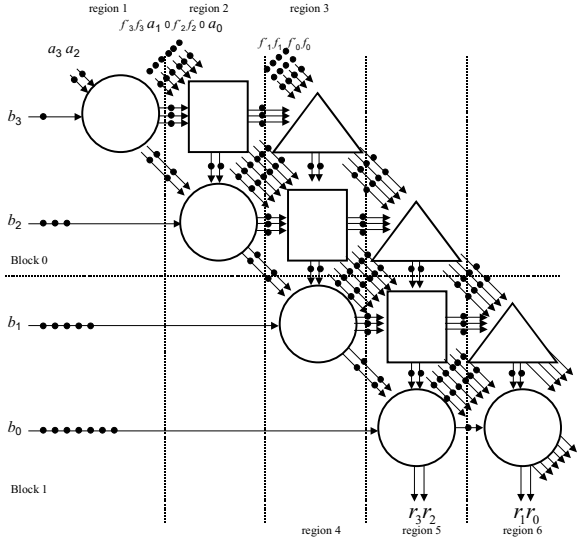
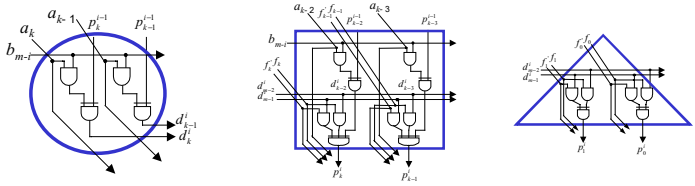


Fig. 3. MSPM in GF(2^4)

The MSPM consists of $(m^2+2m-2)/2$ PE_A cells, that are composed of $(3m-2)/2$ PE_A cells, $(m^2-3m+2)/2$ PE_B cells, and m-1 PE_C cells shown in Figure 4, respectively.



(a) PE_A (b) PE_B (c) PE_C

Fig. 4. Circuit of PEs in Figure 3.

The proposed digit-serial structure is derived from the MSPM. The DG in Figure 3 is partitioned into m/L blocks, where L is a multiple of 2, m/L is an integer, and each block consists of L rows × (m+2L) columns, except that the last block consists of L rows × (m+2(L-1)) columns. Next, each block of the DG is partitioned into $\lceil (m+2L)/L \rceil$ regions. In each block, the first region contains $L/2$ PE_A cells and $\sum_{i=1}^{2i<L}(L/2-i)$ PE_B cells, the second region contains $L/2$ PE_A cells and $((L/2)^2 + \sum_{i=1}^{2i<L}(L/2-i))$ PE_B cells, the second last region contains $L/2$ PE_C cells and $((L/2)^2 + \sum_{i=1}^{2i<L}(L/2-i))$ PE_B cells, the last region contains $L/2$ PE_C cells and $\sum_{i=1}^{2i<L}(L/2-i)$ PE_B cells, and the remaining regions contain $L^2/2$ PE_B cells.

By projecting the DG of Figure 3 along the horizontal direction following the projection procedure in [9], a one-dimensional digit-serial systolic AB^2 multiplier is created, denoted by DSPM, as shown in Figure 5. This array consists of N-1 basic cells, as shown in Figure 6 and 1 basic cell, as shown in Figure 7.

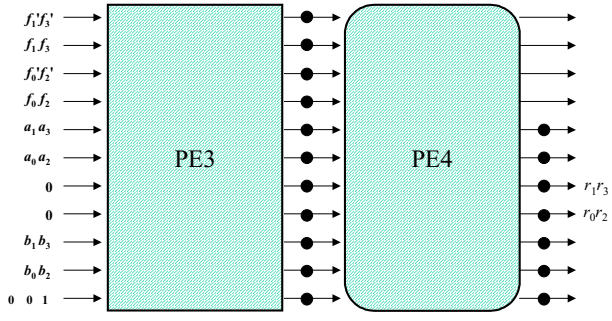


Fig. 5. DSPM in GF(2^4) when L=2

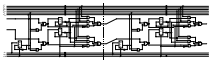


Fig. 6. Circuit for PE3 in Figure 5.

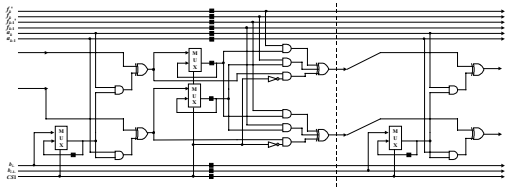


Fig. 7. Circuit for PE4 in Figure 5.

The array is controlled by a control sequence of 1000···0 with length N. The coefficient of the result $r_i s$ emerges from the right-hand side of the array at a rate of L-bits per clock cycle. Since the L temporary results, p_is and b_is, must be broadcast to all the cells in the ith row in Figure 5, 3L multiplexers and 3L one-bit latches are added to Figure 6 and an extra 1+3(L-1) multiplexers and 1+3(L-1) one-bit latches are added to Figure 7. When the control signal is in logic 1, the L temporary results and the value of b are latched. In this case, L^2 two-input AND gates and L^2 NOT gates are added to Figure 6 and an extra L(L-1) two-input AND gates and L(L-1) NOT gates are added to Figure 7.

For the digit-serial systolic array in Figure 5, the maximum propagation delay is $T_{max} = L\ (T_{AND2}+T_{NOT}+T_{XOR2}+T_{XOR3}+T_{MUX})$, where T_{ANDi}, T_{XORi}, T_{NOT}, and T_{MUX} denote the propagation delays through an i-input AND gate, i-input XOR gate, NOT gate, and 2-to-1 multiplexer, respectively. When the digit size L becomes large, the maximum propagation delay also becomes large, thereby decreasing the clock rate. Therefore, to counter such a problem, each basic cell is further pipelined to maintain a small maximum propagation delay when the digit size L becomes large. As such, a high clock rate can be maintained even when the digit size becomes large.

By applying the techniques of the cut theorem [10], the basic cells in Figure 6 and Figure 7 can be easily pipelined in two stages by placing one extra one-bit latch on each of the communication links crossed by dashed lines. For example, with the addition of an extra 5L+1 1-bit latches to Figure 6 and Figure 7, the latency of the array becomes (5m-4)/2 clock cycles, and the maximum propagation delay is reduced to $T'_{max} = T_{AND2}+T_{NOT}+T_{XOR2}+T_{XOR3}+T_{MUX}$.

Therefore, the Area-Time complexity of the pipelined-DSPM is approximately 10.9% lower than that of the non-pipelined DSPM-1, when m=160 and L=8.

4 Systolic *A*/*B* Architecuture

According to the division algorithm based on the binary method [11], Figure 8 shows a systolic architecture of divider for GF(2^4), which uses the (m-1)MSPMs. It consists of $\lceil m/L \rceil(m-1)$ cells and can produce all the result after $(m-1)(\lceil (m+2(L-1))/L \rceil+3(\lceil m/L \rceil-2))$ clock cycles.

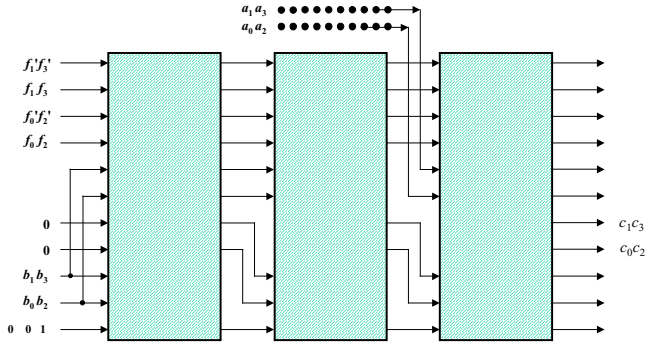


Fig. 8. Digit-serial systolic array for A/B in GF(2^4)

Table 1. Comparison of AB^2 architectures in GF(2^m)

Item \ Circuit	Wang et al [8]	DSPM	Pipelined-DSPM
Architecture	Systolic	Systolic	Systolic
I/O	Bit-parallel	Digit-serial	Digit-serial
No. of cells	$m^2/2$	$\lceil m/L \rceil$	$\lceil m/L \rceil$
Function	$AB^2 + C$	AB^2	AB^2
Throughput	1	L/m	L/m
Critical path	$T_{AND2}+3T_{XOR}$	$L\,(T_{AND2}+T_{NOT}+T_{XOR2}+T_{XOR3}+T_{MUX})$	$T_{AND2}+T_{NOT}+T_{XOR2}+T_{XOR3}+T_{MUX}$
Latency	$2m+m/2$	$\lceil (m+2(L-1))/L \rceil+3(m/L-1)$	$(5m-4)/2$
AND gates	$3m^2$	$\lceil m/L \rceil 4L^2-3$	$\lceil m/L \rceil 4L^2-3$
XOR gates	$3m^2$	$\lceil m/L \rceil 3L^2-2$	$\lceil m/L \rceil 3L^2-2$
Latches	$8.5m^2$	$\lceil m/L \rceil (3L^2+5L)-2$	$\lceil m/L \rceil (7L^2+5L-3)-4L$
Mux		$\lceil m/L \rceil 3L-2$	$\lceil m/L \rceil 3L-2$
NOT gates		$\lceil m/L \rceil L^2-L$	$\lceil m/L \rceil L^2-L$
No. of CS	-	1	1

Table 2. Comparison of A/B architectures in GF(2^m)

Item \ Circuit	Wang [8][Fig.4]	Proposed divider
Architecture	Systolic	Systolic
I/O format	Bit-parallel	Digit-serial
Number of cells	$m^2(m-1)/2$	$\lceil m/L \rceil (m-1)$
Function	A/B	A/B
Throughput	1	L/m
Critical Path	$T_{AND2}+T_{XOR4}$	$T_{AND}+T_{XOR3}+T_{XOR2}+T_{MUX}+T_{NOT}$
Latency	$2m^2-3m/2$	$(m-1)(\lceil (m+2(L-1))/L \rceil+3(\lceil m/L \rceil-2))$
AND gates	$3m^3-3m^2$	$\lceil m^2/L-m/L \rceil (4L^2-3)$
XOR gates	$3m^3-3m^2$	$\lceil m^2/L-m/L \rceil (3L^2-2)$
Latches	$8.5m^3-8.5m^2$	$\lceil m^2/L-m/L \rceil (3L^2+5L)-(3L^2+L-2)$
Not gates		$\lceil m^2/L-m/L \rceil (L^2-L)$
Mux		$\lceil m^2/L-m/L \rceil (3L-2)$
No. of CS	-	1

In the digit-serial divider, when the digit-size, L, was selected to be less than about (1/2)m, the proposed digit-serial divider was more efficient than Wang[8]'s bit-parallel divider, which is based on the area-time product [12]. Therefore, if the appropriate digit-size is selected, the digit-serial arrays were more efficient than bit-parallel architectures in terms of the area and time complexity.

5 Conclusion

This paper presented digit-serial-in-digit-serial-out systolic AB^2 and A/B architecture in GF(2^m). Table 1 and Table 2 show comparisons of the proposed digit-serial systolic architectures with the. In the table 1, the latency of the power-sum circuit in [8] was $2m+m/2$ cycles, while the proposed DSPM had $\lceil (m+2(L-1))/L \rceil+3(m/L-1)$ cycles when m=160 and L=2, representing a latency reduction of approximately 20% compared to [8]. Furthermore, the proposed architecture also allows the digit-size of the regular square form to be selected.

When we compares the proposed architectures with those developed by Wang [8] based on the area-time product [12]. The results showed that the proposed arrays were more efficient in terms of the area and time complexity. That is, when the digit-size, L, was selected to be less than m, the proposed DSPM architecture was more efficient than Wang[8]'s architecture.

References

[1] D.E.R.Denning, *Cryptography and data security*, Addison-Wesley, MA, 1983.
[2] A.Menezes, *Elliptic Curve Public Key Cryptosystems*, Kluwer Academic Publishers, Boston, 1993.
[3] S.W. Wei, "A Systolic Power-Sum Circuit for GF(2m)," *IEEE Trans. Computers*, 43: 226-229, 1994.
[4] J.H.Guo, C.L.Wang, 'Digit-serial systolic multiplier for finite fields GF(2m),' *IEE Proc.-Comput. Digit. Tech*., Vol.145, 1998.
[5] C. L. Wang and J. H. Guo, 'New systolic arrays for $C+AB2$, inversion, and division in GF(2m),' *IEEE Transactions on Computers*, Vol. 49, No. 10, pp. 1120-1125, 2000.
[6] S.W. Wei, "A Systolic Power-Sum Circuit for GF(2m)," *IEEE Trans. Computers*, 43: 226-229, 1994.
[7] J.V.McCanny, R.A.Evans and J.G.Mcwhirter, 'Use of unidirectional data flow in bit-level systolic array chips', *Electron. Lett*., 22, pp.540-541, 1986.
[8] C.L.Wang and J.H.Guo, 'New systolic arrays for $C+AB2$, inversion, and division in GF(2m),' *IEEE Trans. Computers*, 29, pp. 1120-1125, 2000.
[9] S.Y.Kung, *VLSI array processors*, Prentice Hall, Englewood Cliffs, NJ, 1988.
[10] Kung, H.T., and LAM, M., 'Fault tolerant and two level pipelining in VLSI systolic arrays,' MIT conference on *Advanced res. VLSI*, Cambridge, MA, January 1984, pp.74-83.
[11] D.E.Knuth, *The Art of Computer Programming, volume 2: Seminumerical Algorithms*, Addison-Wesley, Reading, Massachusetts, 2nd edition, 1997.
[12] Daniel D. Gajski, *Principles of Digital Design*, Prentice-hall international, INC, 1997.

Design and Experiment of a Communication-Aware Parallel Quicksort with Weighted Partition of Processors

Sangman Moh[1], Chansu Yu[2], and Dongsoo Han[3]

[1] Dept. of Internet Eng., Chosun Univ.
375 Seoseok-dong, Dong-gu, Gwangju, 501-759 KOREA
smmoh@chosun.ac.kr
[2] Dept. of Electrical and Computer Eng., Cleveland State Univ.
Cleveland, OH 44115, USA
c.yu91@csuohio.edu
[3] School of Eng., Information and Communications Univ.
58-4 Hwaam-dong, Yuseong-gu, Daejeon, 305-348 KOREA
dshan@icu.ac.kr

Abstract. In most parallel algorithms, inter-processor *communication cost* is much more than *computing cost* within a processor. So, it is very important to reduce the amount of inter-processor communication. This paper presents the design and experiment of a new communication-aware parallel quicksort scheme for distributed-memory multiprocessor systems. The key idea of the proposed scheme is the *weighted partition* of processors, which enables not only less inter-processor communication but also better load balancing among the participating processors during the quicksort. The proposed scheme was designed and experimented on the Cray T3E parallel computer. According to the comparative performance measurement, for up to 64 processors, the proposed scheme results in about 40 ~ 60 percent shorter run time compared to the conventional parallel quicksort. That is mainly due to the small amount of inter-processor communication that results from the weighted partition and allocation of processors. The performance improvement is more substantial as the number of processors, the input size, and the input item size increases.

1 Introduction

Sorting is a fundamental operation that appears in many computing applications; it rearranges a list of input numbers in non-decreasing (or non-increasing) order. In any sequential sorting, the best performance is bounded to O(n log n) which is achieved by two well-known algorithms: mergesort and quicksort [1]. *Quicksort* [2-3] is often the best practical choice for sorting because it is remarkably efficient on the average and the constant factors hidden in the O(n log n) notation are quite small. The best-case performance of quicksort is O(n log n) and it is proven to be the same as the average performance while the worst-case performance of quicksort is O(n^2) [1].

Since O(n log n) is optimal for any sequential sorting algorithm that does not use any special properties for the input patterns, the best parallel time complexity we can expect for a sequential algorithm using n processors is O(n log n) / n = O(log n).

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 97–105, 2004.
© Springer-Verlag Berlin Heidelberg 2004

Leighton [4] demonstrated an O(log *n*) sorting algorithm with *n* processors based on an algorithm by Ajtai, Komlos, and Szemeredi [5], but the constant hidden in the order notation was extremely large. Bitton et al. [6] published an extensive survey paper on parallel sorting. Akl [7] wrote a book devoted entirely to parallel sorting algorithms, which describes 20 different parallel sorting algorithms. The outcome of all this investigation is that a realistic O(log *n*) algorithm with *n* processors is a goal that will not be easy to achieve [8].

For parallel computer systems, some parallel sorting algorithms have been newly developed. On the other hand, the parallelized version of the sequential sorting algorithms has been also researched and used more actively than the newly developed parallel sorting algorithms [8]. We also focus on the *parallelized algorithm* of sequential sorting. In particular, our work concentrates on quicksort, which is popular and effectively used in many computing areas. It has been implemented on several well-known architectures such as hypercubes [9-10]. Jelenkovic and Omecen-Ceko [11] presented some experiments with multithreading in parallel quicksort. In order to speed up the computation-intensive tasks of sorting, a dedicated hardware solution was researched [12].

In most parallel algorithms, inter-processor *communication cost* is much more than *computing cost* within a processor. So, it is very important to reduce the amount of inter-processor communication. This paper proposes a *communication-aware parallel quicksort* scheme that is suitable for distributed-memory multiprocessor systems. The key idea of the proposed scheme is the *weighted partition* of processors, which enables not only less inter-processor communication but also better load balancing among the participating processors during the quicksort. We implemented the proposed scheme in C language using *MPI* APIs and ran it on the Cray T3E parallel computer. We then measured the performance of the proposed scheme and compared it with that of the conventional parallel quicksort [8]. According to our extensive performance measurement, for up to 64 processors, the proposed scheme results in about 40 ~ 60 percent shorter run time than the conventional scheme. This improvement is primarily due to the small amount of inter-processor communication that results from the weighted partition and allocation of processors, compared to the conventional approach. The performance improvement is more substantial as the number of processors, the input size, and the input item size increases. In addition, a more balanced partition of input numbers to participating processors is achieved.

The rest of the paper is organized as follows: Conventional parallel quicksort is reviewed in the following section. Section 3 presents the proposed communication-aware parallel quicksort scheme with examples. Experiment and performance results are discussed in Section 4. Finally, conclusion is covered in Section 5.

2 Related Work

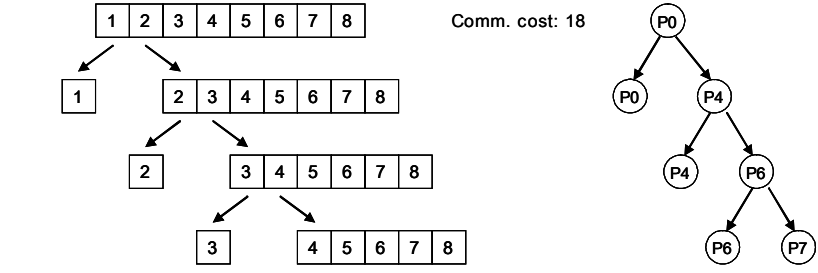
Quicksort divides a list of input numbers into two sublists by choosing a *pivot* and moving the numbers smaller than the pivot into one list and the larger numbers into the other list. The algorithm then recursively sorts the sublists by choosing a new

pivot and subdividing each of the sublists. If an input number is smaller than the pivot, it is placed in the left sublist. Otherwise, it is placed in the right sublist. The pivot could be any input number in the list, but often the first number in the list is chosen.

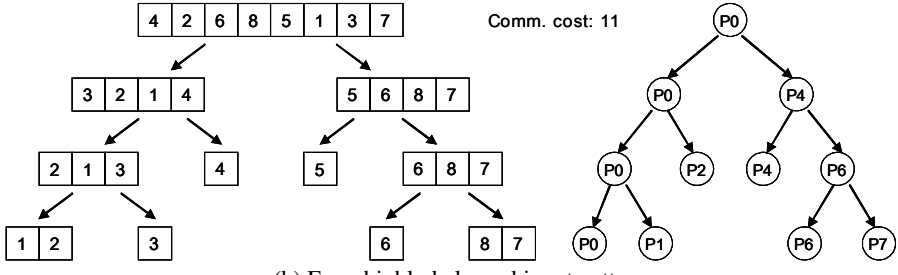
Quicksort is based on the *divide-and-conquer* concept, which consists of partitioning and merging. The partitioning is the major time-consuming part of quicksort, whereas the merging phase is very simple. The procedure is repeated on the partitioned sublists recursively. By repeating the procedure recursively, we are left with sublists of one number each. With proper merging (combining) of the sublists, a sorted list is obtained.

The code of quicksort can be formed as follows:

```
quicksort(list, start, end)
  {
     if (start < end) {
       partition(list,start,end,pivot);
       quicksort(list, start, pivot-1);
       quicksort(list, pivot+1, end);
     }
  }
```



(a) For a worst-case input pattern



(b) For a highly balanced input pattern

Fig. 1. Examples of the conventional parallel quicksort.

The function `partition()` moves numbers in the list between `start` to `end` so that those less than the pivot are before the pivot and those equal to or greater than the pivot are after the pivot.

One obvious way to parallelize quicksort is to start with one processor and pass on one of the recursive calls to another processor while keeping the other recursive call to perform. In the tree structure of parallel quicksort, the pivot is carried with the left list until the final sorting action. The conventional parallel quicksort algorithm is well described in [8], and two examples of this algorithm are shown in Fig. 1.

As Fig. 1 reveals, in general, the tree structure in quicksort may not be perfectly balanced. The sort tree becomes unbalanced if the pivots do not divide the lists into equal sublists. When we choose the first number in a sublist as the pivot, the original ordering of the numbers being sorted is the determining factor in the speed of the quicksort.

3 Design of a Communication-Aware Parallel Quicksort

As mentioned earlier, the key idea of the proposed scheme is the weighted partition and allocation of processors, which enables not only less inter-processor communication but also better load balancing among the participating processors during the quicksort. Initially, the master processor takes the input list. By default, the master processor has the lowest processor identifier (*i.e.,* P_0) among the participating processors.

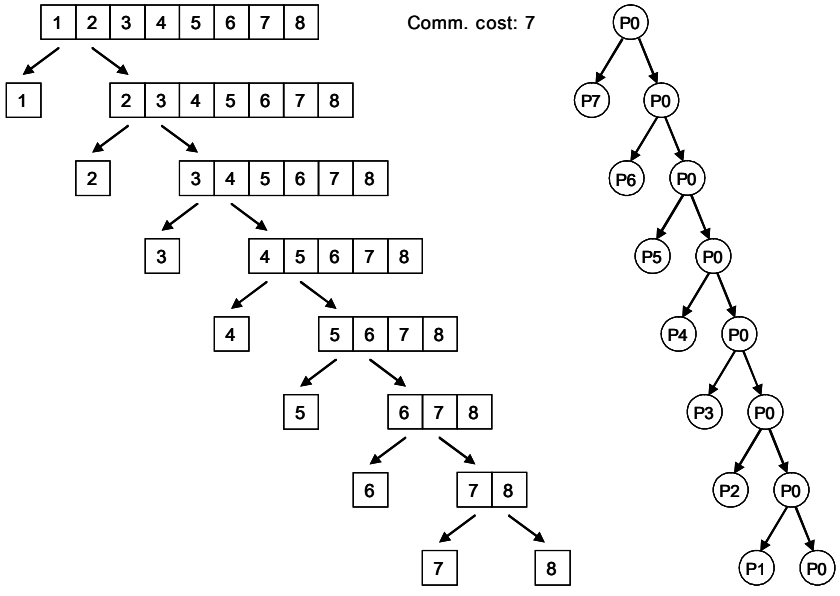
Let the number of input items that are less than the pivot and the number of input items that are greater than the pivot be N_L and N_R, respectively. Let the partition composed of input items that are less than the pivot and the partition composed of input items that are greater than the pivot be P_L and P_R, respectively. Then, at each level of *recursive partitioning* (tree operation), the proposed parallel quicksort operates as follows:

(1) Partition the processors into two subpartitions by the ratio of N_L to N_R in the non-decreasing order of processor identifiers;

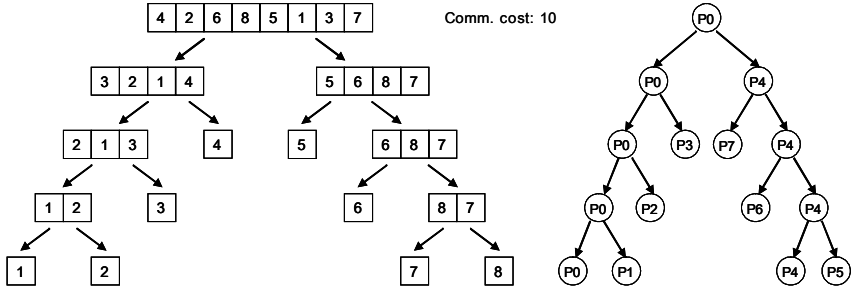
(2) Send the smaller of P_L and P_R to the first processor in the other subpartition without the current processor.

The rest parts of the proposed scheme are the same as the conventional parallel quicksort [8], that is, the proposed scheme partitions the participating processors into two groups by the ratio of the size of two sublists and assigns (sends) the smaller of the two sublists to the other group without the current processor. We implemented the proposed scheme in C language using MPI APIs on the Cray T3E parallel computer, and the experiment results are discussed in Section 4.

Given an input list and processors, the proposed scheme minimizes the amount of inter-processor communication. As Fig. 2(a) shows, in the worst case, this scheme remarkably reduces the number of messages transferred between processors. Moreover, due to the weighted partition and allocation of processors, the communication cost is reduced and the parallelized computation is more balanced among participating processors. It results in the shorter run time of the proposed parallel quicksort compared to the conventional one.



(a) For a worst-case input pattern



(b) For a highly balanced input pattern

Fig. 2. Examples of the proposed communication-aware parallel quicksort.

Fig. 2 reveals the following: (i) for the worst-case input patterns, the proposed scheme outperforms the conventional one, (ii) for the best-case input patterns, the proposed scheme has the same performance as the conventional one, and (iii) for most of general input patterns, the proposed scheme also outperforms the conventional one. Thus, we can conclude that our approach is better than the conventional parallel quicksort.

For a worst-case input pattern, in the conventional parallel quicksort in Fig.1, the inter-processor communication cost is 18 and four out of eight processors are effectively used during the sorting, where the inter-processor communication cost represents the normalized amount of data transferred among processors. On the other hand, in the proposed parallel quicksort, the inter-processor communication cost is 7 and all the eight processors are effectively used, resulting in better performance.

4 Experiment and Performance Evaluation

In order to evaluate the performance of the proposed scheme and compare it with that of the conventional scheme, we implemented and ran both schemes on the Cray T3E parallel computer system in C language using MPI APIs. We then measured the run time of the two parallel quicksort schemes.

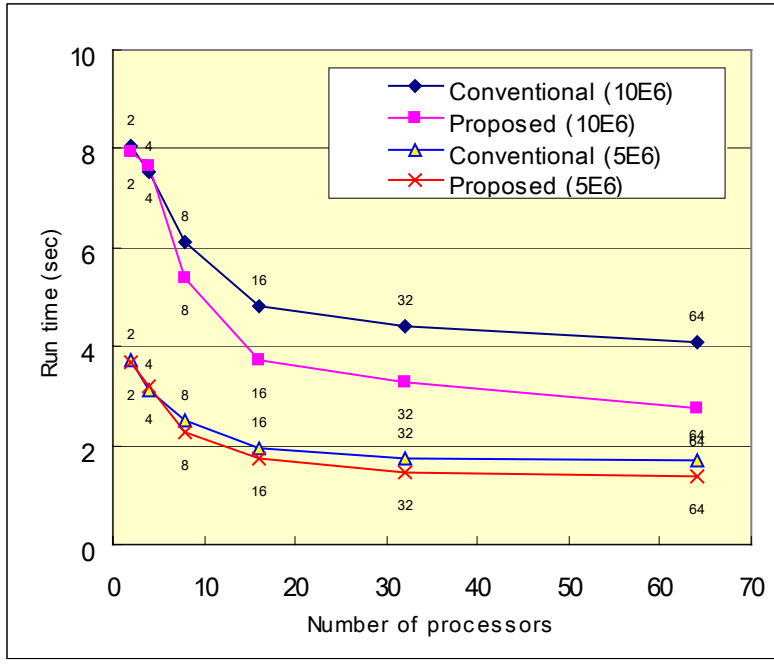


Fig. 3. Performance of parallel quicksort schemes (input item size = 4 bytes).

In our practical measurement, the input patterns were randomly generated and then the execution time was measured by an in-line timing check function inserted into the quicksort programs. Since both quicksort schemes used randomized input patterns, we can conclude that a reasonable average performance was obtained in our measurements. Note here that, during the run time, the inter-processor communication cost could not be measured separately from run time; however, it is inherently included in the run time.

Fig. 3 shows the execution time of the two parallel quicksort schemes, which were measured for input sizes of 5,000,000 and 10,000,000, where each input item is 4 bytes long. As the figure shows, the proposed parallel quicksort sorts the same size problem in shorter time than the conventional parallel quicksort. For instance, for 64 processors, the proposed scheme is faster than the conventional scheme by factors of 1.35 and 1.50 for the input sizes of 5,000,000 and 10,000,000, respectively. When few processors (up to 4) are used, the performance gain is small, because the difference between the two schemes is negligible for a small number of processors.

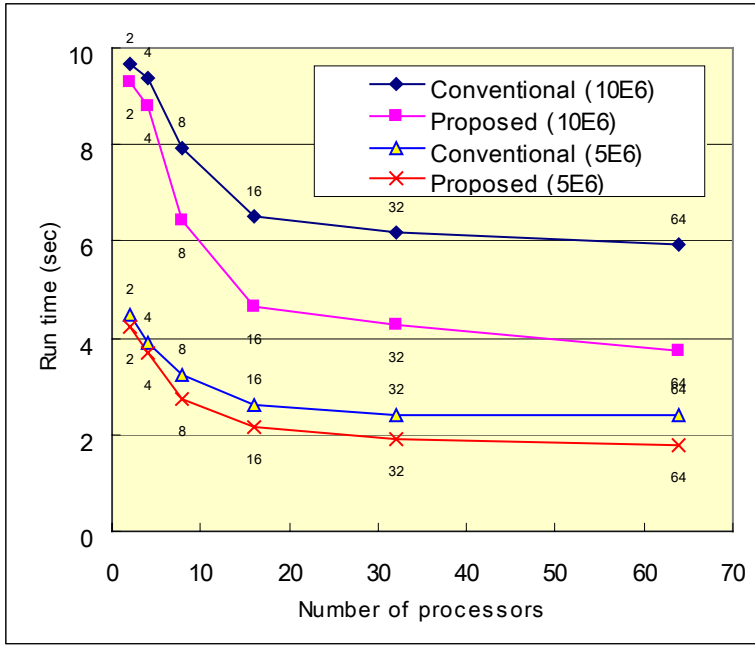


Fig. 4. Performance of parallel quicksort schemes (input item size = 8 bytes).

Fig. 4 shows the same performance metric as depicted in Fig. 3 except that each input item is 8 bytes long. In this case, for 64 processors, the proposed scheme is faster than the conventional one by factors of 1.43 and 1.61 for the input sizes of 5,000,000 and 10,000,000, respectively. From Fig. 3 and 4, it is clear that the per-

formance is better as the input item size increases. This is mainly due to the fact that the communication cost increases as the input item size increases but the proposed scheme is more communication-efficient than the conventional one. Conclusively, the performance improvement is more substantial (i) as the number of processors increases, (ii) as the input size increases, and (iii) the input item size increases.

5 Conclusion

In this paper, a new communication-aware parallel quicksort scheme has been presented and discussed, which was implemented on the Cray T3E parallel computer in C language using MPI APIs. The key idea of the proposed scheme is the *weighted partition* of processors, which enables not only less inter-processor communication but also better load balancing among the participating processors during the quicksort. According to the extensive experiment results, the proposed scheme reduces the sorting time by 40 ~ 60 percent for up to 64 processors and is more communication efficient than the conventional scheme. The performance improvement is more substantial as the number of processors, the input size, and the input item size increases. This effect is mainly due to the weighted partition and allocation of processors. In addition, a more balanced partition of input numbers to participating processors is achieved.

In the near future, the proposed scheme will be implemented on a Linux-based PC cluster system using the MPI interface. In cluster systems, since most interconnection networks (*i.e.,* clustering networks) are much slower than the dedicated proprietary interconnection networks used in massively parallel multicomputers, it can be easily inferred that the performance gain is more improved.

References

1. Cormen, T.H., Leiserson, C.E., and Rivest, R.L.: Introduction to Algorithms, MIT Press, Cambridge, Massachusetts (1994)
2. Hoare, C.A.R.: Quicksort, *Computer Journal*, Vol. 5 (1962) 10-15
3. Wainwright, R.L.: A Class of Sorting Algorithms Based on Quicksort, Comm. of ACM, Vol. 28 (1985) 396-402
4. Leighton, F.T.: Tight Bounds on the Complexity of Parallel Sorting, Proc. 16th Annual ACM Symp. on Theory of Computing, New York (1984) 71-80
5. Ajtai, M., Komlos, J., and Szemeredi, E.: An O(n log n) Sorting Network, Proc. 15th Annual SCM Symp. on Theory of Computing, Boston, Massachusetts (1983) 1-9
6. Bitton, D., DeWitt, D.J., Hsiao, D.K., and Menon, J.: A Taxonomy of Parallel Sorting, Computing Surveys, Vol. 16 (1984) 287-318
7. Akl, S.: Parallel Sorting Algorithms, Academic Press, New York (1985)
8. Wilkinson B. and Allen, M.: Sorting Algorithms, Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers, Prentice-Hall, Upper Saddle River, New Jersey (1999)267-297
9. Fox, G.C., Williams, R.D., and Messina, P.C.: Parallel Computing Works, Morgan Kaufmann, San Francisco, California (1994)

10. Quinn, M.J.: Parallel Computing: Theory and Practice, 2nd Ed., McGraw-Hill, New York (1994)
11. Jelenkovic L. and Omecen-Ceko, G.: Experiments with Multithreading in Parallel Computing, Proc. 19 Int. Conf. on Information Technology Intercafes, Pula, Croatia (1997) 357-362
12. Beyer, D.A.: Memory Optimization for a Parallel Sorting Hardware Architecture, Thesis of MS, Electrical and Computer Engineering, Oregon State University (1998)

A Linear Systolic Array for Multiplication in $GF(2^m)$ for High Speed Cryptographic Processors

Soonhak Kwon[1], Chang Hoon Kim[2], and Chun Pyo Hong[2]

[1] Inst. of Basic Science and Dept. of Mathematics, Sungkyunkwan University, Suwon 440-746, Korea
shkwon@math.skku.ac.kr

[2] Dept. of Computer and Information Engineering, Daegu University, Kyungsan 712-714, Korea
chkim@dsp.taegu.ac.kr,cphong@daegu.ac.kr

Abstract. We present new designs of low complexity and low latency systolic arrays for multiplication in $GF(2^m)$ when there is an irreducible all one polynomial (AOP) of degree m. Our proposed bit parallel array has a reduced latency and hardware complexity compared with previously proposed designs. For a cryptographic purpose, we derive a linear systolic array using our algorithm and show that our design has a latency $m/2+1$ and a throughput rate $1/(m/2+1)$. Compared with other linear systolic arrays, we find that our design has at least 50 percent reduced hardware complexity and latency, and has twice higher throughput rate. Therefore our multiplier provides a fast and a hardware efficient architecture for multiplication of two elements in $GF(2^m)$ for large m.

Keywords: Finite field multiplier, systolic array, all one polynomial, Riemann Hypothesis, Artin's conjecture for primitive roots.

1 Introduction

Arithmetic of finite fields, especially finite field multiplication, is very important in many cryptographic areas. Therefore an efficient design of a finite field multiplier is needed. A good multiplication algorithm depends on the choice of a basis for a given finite field. In general, there are three types of basis being used, that is, polynomial, dual and normal basis. Some popular multipliers for various purposes are Berlekamp type dual basis multipliers [1] and Massey-Omura type normal basis multipliers [2,7]. Above mentioned multipliers and other traditional multipliers have some unappealing characteristics. For example, they have irregular circuit designs. In other words, their hardware structures may be quite different for varying choices of m for $GF(2^m)$, though the multiplication algorithm is basically same for each m. Moreover as m gets large, the propagation delay also increases. So deterioration of the performance is inevitable.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 106–116, 2004.
© Springer-Verlag Berlin Heidelberg 2004

Systolic multipliers [3,4,5,6] do not suffer from above problems. They have regular structures consisting of a number of replicated basic cells, each of which has the same circuit design. So overall architectures of systolic multipliers are the same and do not depend on a particular choice of m for $GF(2^m)$. Furthermore since each basic cell is only connected with its neighboring cells, signals can be propagated at a high clock speed. Recently, Lee et al. [6] proposed a very efficient bit parallel systolic array using an extended all one polynomial (AOP) basis. This multiplier has a low cell complexity and a high throughput rate when compared with other multipliers.

In this paper, we present an improved design of a bit parallel systolic array in [6]. We show that the hardware complexity and the latency of our bit parallel systolic multiplier are significantly reduced when compared with the design in [6]. Also, we propose a linear systolic array by modifying our multiplication algorithm. It is shown that our linear array has the reduced latency and hardware complexity by at least 50 percent compared with other existing linear systolic arrays. Moreover since our linear array has a high throughput rate, it can be used in many cryptographic applications.

2 All One Polynomial and a Systolic Array for Multiplication in $GF(2^m)$

Let $GF(2^m)$ be a finite field with 2^m elements. Define a polynomial of degree m, $f(X) = \frac{X^{m+1}-1}{X-1} = 1 + X + X^2 + \cdots + X^m \in GF(2)[X]$. It is called an all one polynomial (AOP). One can easily show that $f(X)$ is irreducible over $GF(2)$ if and only if $m+1 = p$ is a prime and 2 is a primitive root (mod p). Letting $\alpha \in GF(2^m)$ be any zero of the $f(X)$, we have a polynomial basis $\{1, \alpha, \alpha^2, \cdots, \alpha^{m-1}\}$ for $GF(2^m)$ over $GF(2)$. For any $x \in GF(2^m)$ with an irreducible AOP of degree m, one may write x as $x = \sum_{i=0}^{m} x_i \alpha^i$ with respect to the extended AOP basis $\{1, \alpha, \alpha^2, \cdots, \alpha^m\}$. The extended AOP basis is not really a basis because of the redundancy (linear dependence) of the basis elements. However in many situations, by using the nice property $\alpha^{m+1} = 1$, we can get an area efficient multiplier over $GF(2^m)$. For example, a bit serial multiplier of Berlekamp type using an extended AOP basis is presented in [8]. It is known [15] that the number of $m \leq 2000$ for which an AOP basis exists is 118. For example, we have an AOP basis when $m = 2, 4, 10, 12, 18, 28, 36, 52, 58, 60, 66, 82, 100, 106, \cdots$. From now on, we assume that there is an irreducible AOP, $f(X) = 1 + X + X^2 + \cdots + X^m$, of degree m with $f(\alpha) = 0$.

Definition 1. *Let $x = \sum_{i=0}^{m} x_i \alpha^i$ be an element in $GF(2^m)$. Define $x_i \in GF(2)$ for all integers i as $x_i = x_j$ if $i \equiv j \pmod{m+1}$ for some j in $\{0, 1, 2, \cdots, m\}$.*

Therefore we can talk about the coefficients x_i of $x \in GF(2^m)$ for $i < 0$ and for $i > m$. From this definition and using $\alpha^{m+1} = 1$, we easily get

Lemma 1. *Let* $x = \sum_{i=0}^{m} x_i\alpha^i, y = \sum_{i=0}^{m} y_i\alpha^i$ *be two elements in* $GF(2^m)$. *Then we have* $xy = \sum_{k=0}^{m}(xy)_k\alpha^k$ *where the kth coefficient* $(xy)_k$ *is written as* $(xy)_k = \sum_{i=0}^{m} y_i x_{k-i}$.

The above lemma is well known [6,8] in many different notations and the proof is trivial once we notice

$$xy = \sum_{i=0}^{m} x_i\alpha^i \sum_{j=0}^{m} y_j\alpha^j = \sum x_i y_j \alpha^{i+j} = \sum_{i=0}^{m}\sum_{j=0}^{m} y_j x_{\langle i-j\rangle}\alpha^i,$$

where $\langle i-j\rangle$ is a unique integer in $\{0,1,2,\cdots,m\}$ satisfying $\langle i-j\rangle \equiv i-j \pmod{m+1}$. The advantage of Definition 1 in the expression of Lemma 1 without the notation $\langle\ \rangle$ will be explained soon. Note that, since $m+1=p$ is an odd prime, $\{0,2,4,\cdots,2m\}$ and $\{0,1,2,\cdots,m\}$ are same sets modulo $m+1$. Thus for any $x \in GF(2^m)$, using Definition 1, we have $\{x_0,x_2,x_4,\cdots,x_{2m}\} = \{x_0,x_1,x_2,\cdots,x_m\}$. Now from Lemma 1, we have the kth coefficient $(xy)_k$ of xy in $GF(2^m)$ as a matrix multiplication of a row vector and a column vector,

$$(xy)_k = \sum_{i=0}^{m} y_i x_{k-i} = (x_k, x_{k-1}, \cdots, x_{k-m})(y_0, y_1, \cdots, y_m)^T,$$

where $(y_0,y_1,\cdots,y_m)^T$ is the transposition of the row vector $(y_0,y_1,\cdots,y_m)$. From this information, we may derive the following result which was already discovered in Lee et al. [6] with different notations.

Theorem 1. *Let* $x = \sum_{i=0}^{m} x_i\alpha^i, y = \sum_{i=0}^{m} y_i\alpha^i$ *be two elements in* $GF(2^m)$. *Then, for any integer k, we have* $(xy)_{2k} = \sum_{i=0}^{m} y_{k+i}x_{k-i}$.

Proof. By using Definition 1 and Lemma 1,

$$\begin{aligned}(xy)_{2k} &= \sum_{i=0}^{m} y_i x_{2k-i} = (x_{2k}, x_{2k-1}, \cdots, x_{2k-m})(y_0, y_1, \cdots, y_m)^T \\ &= (x_k, x_{k-1}, \cdots, x_{k-m})(y_k, y_{k+1}, \cdots, y_{k+m})^T = \sum_{i=0}^{m} y_{k+i}x_{k-i},\end{aligned}$$

where the fourth expression is obtained by shifting the vectors in the third expression k positions to the left. □

In [6], though the idea is very original and illuminating, they used a rather complicated argument, to derive above theorem and the corresponding systolic array, by introducing some unconventional definition of an inner product of two elements in $GF(2^m)$.

However our explanation in Theorem 1 is simple and easy to understand. It is obvious from Theorem 1 that the basic cell of the bit parallel systolic array can be described as in Fig. 1, where • denotes one bit latch (flip-flop). We omit the corresponding systolic array since it is exactly same to the design in [6]. Notice that the basic cell needs three latches and the number of cells is $(m+1)^2$. Consequently the latency of the multiplier in [6] is $m+1$, while other bit parallel systolic arrays [3,4,5] have latency $3m$.

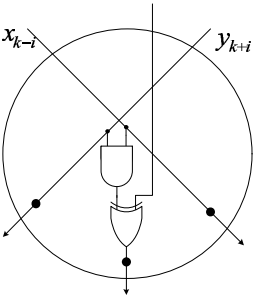


Fig. 1. The circuit of (i, k) basic cell in [6]

3 Improved Design of a Bit Parallel Systolic Array Using Irreducible AOP

By modifying Theorem 1, we may further reduce the latency and the hardware complexity of the bit parallel systolic multiplier presented in [6]. This is explained in the following theorem.

Theorem 2. *Let* $x = \sum_{i=0}^{m} x_i\alpha^i, y = \sum_{i=0}^{m} y_i\alpha^i$ *be two elements in* $GF(2^m)$. *Then, for any integer* k, *we have* $(xy)_{2k} = \sum_{i=0}^{m/2-1}(y_{k+i}x_{k-i} + y_{k-i-1}x_{k+i+1}) + y_{k+m/2}x_{k-m/2}$.

Proof. Using Theorem 1, we have

$$\begin{aligned}(xy)_{2k} &= \sum_{i=0}^{m} y_{k+i}x_{k-i} = \sum_{i=0}^{m/2-1} y_{k+i}x_{k-i} + \sum_{i=m/2+1}^{m} y_{k+i}x_{k-i} + y_{k+m/2}x_{k-m/2} \\ &= \sum_{i=0}^{m/2-1} y_{k+i}x_{k-i} + \sum_{i=0}^{m/2-1} y_{k+m-i}x_{k-(m-i)} + y_{k+m/2}x_{k-m/2} \\ &= \sum_{i=0}^{m/2-1} y_{k+i}x_{k-i} + \sum_{i=0}^{m/2-1} y_{k-i-1}x_{k+i+1} + y_{k+m/2}x_{k-m/2},\end{aligned}$$

where the third equality follows by rearranging the summands and the fourth equality follows from Definition 1. □

Now for each $(xy)_{2k}$, we define a column vector

$$\mathcal{W}_k = (w_{0k}, w_{1k}, \cdots, w_{(m/2-1)k}, w_{(m/2)k})^T,$$

where

$$\begin{aligned} w_{ik} &= y_{k+i}x_{k-i} + y_{k-i-1}x_{k+i+1}, \quad if\ \ 0 \le i \le m/2-1 \\ w_{(m/2)k} &= y_{k+m/2}x_{k-m/2}, \qquad\qquad\quad if\ \ i = m/2. \end{aligned}$$

Then the sum of all entries of the column vector $\mathcal{W}_k$ is exactly $(xy)_{2k}$ and $\mathcal{W}_k$ appears as a kth $(0 \le k \le m)$ column vector of the $m/2+1$ by $m+1$ matrix $\mathcal{W} = (w_{ik})$ where

$$\mathcal{W} = \begin{pmatrix} w_{00} & w_{01} & w_{02} & \cdots & w_{0m} \\ w_{10} & w_{11} & w_{12} & \cdots & w_{1m} \\ w_{20} & w_{21} & w_{22} & \cdots & w_{2m} \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdot & \cdots & \cdot \\ w_{(m/2-1)0} & w_{(m/2-1)1} & w_{(m/2-1)2} & \cdots & w_{(m/2-1)m} \\ w_{(m/2)0} & w_{(m/2)1} & w_{(m/2)2} & \cdots & w_{(m/2)m} \end{pmatrix}.$$

For each $0 \le i \le m/2-1$ and $0 \le k \le m$, using the relation

$$w_{ik} = y_{k+i}x_{k-i} + y_{k-i-1}x_{k+i+1},$$

we have

$$w_{(i-1)(k-1)} = y_{k+i-2}x_{k-i} + y_{k-i-1}x_{k+i-1}.$$

That is, the signals x_{k-i} and y_{k-i-1} in the expression of w_{ik} come from the signals in the expression of $w_{(i-1)(k-1)}$. Also since

$$w_{(i-1)(k+1)} = y_{k+i}x_{k-i+2} + y_{k-i+1}x_{k+i+1},$$

we deduce that the signals x_{k+i+1} and y_{k+i} in the expression of w_{ik} come from the signals in the expression of $w_{(i-1)(k+1)}$. Moreover the signals in the last row come from the signals in the $m/2-1$th row. That is, $w_{(m/2)0} = y_{m/2}x_{-m/2} = y_{m/2}x_{m/2+1}$ comes from the signals $y_{m/2}$ and $x_{m/2+1}$ in the expression $w_{(m/2-1)1} = y_{m/2}x_{2-m/2} + y_{1-m/2}x_{m/2+1}$.

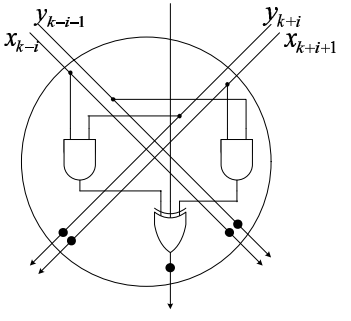


Fig. 2. An improved circuit of (i,k) basic cell

And for each $1 \le k \le m$, $w_{(m/2)k} = y_{k+m/2}x_{k-m/2}$ comes from the signals $y_{k+m/2}$ and $x_{k-m/2}$ in the expression $w_{(m/2-1)(k-1)} = y_{k+m/2-2}x_{k-m/2} + y_{k-m/2-1}x_{k+m/2-1} = y_{k+m/2-2}x_{k-m/2} + y_{k+m/2}x_{k+m/2-1}$. Therefore we may

construct a bit parallel systolic multiplier with respect to the basis $\{1, \alpha^2, \alpha^4, \cdots, \alpha^{2m}\}$. The circuit of basic cell is explained in Fig. 2, where $\bullet$ is one bit latch (flip-flop). For simplicity, we assume $m = 4$. Then the matrix $\mathcal{W}$ is as follows.

$$\mathcal{W} = \begin{pmatrix} y_0x_0 + y_4x_1 & y_1x_1 + y_0x_2 & y_2x_2 + y_1x_3 & y_3x_3 + y_2x_4 & y_4x_4 + y_3x_0 \\ y_1x_4 + y_3x_2 & y_2x_0 + y_4x_3 & y_3x_1 + y_0x_4 & y_4x_2 + y_1x_0 & y_0x_3 + y_2x_1 \\ y_2x_3 & y_3x_4 & y_4x_0 & y_0x_1 & y_1x_2 \end{pmatrix}.$$

Letting $z = \sum_{i=0}^{m} z_i\alpha^i \in GF(2^m)$, we may realize the product sum operation $xy + z$ in the bit parallel systolic arrangement shown in Fig. 3.

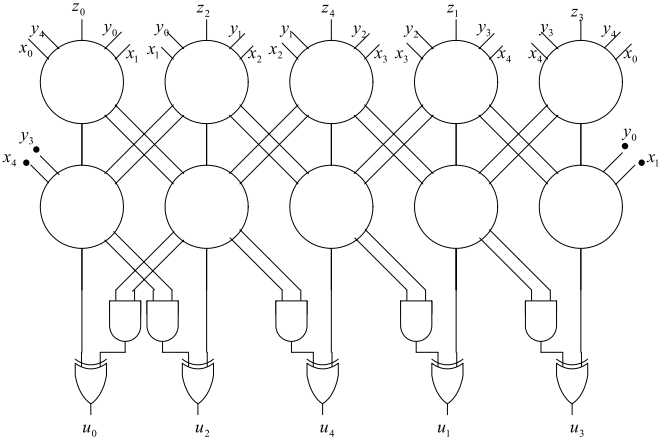


Fig. 3. A new systolic architecture for computing $u = xy + z$ in $GF(2^4)$

We compare our multiplier with other bit parallel systolic arrays in Table 1. In Table 1, AND and XOR mean 2-input gates and 3XOR means a 3-input XOR gate. D_A, D_X, D_{3X} and D_L denote the delay time of an AND, a XOR, a 3XOR and a latch respectively.

Table 1. Comparison of our bit parallel systolic array with other multipliers of the same type.

	Wang [3]	Yeh [4]	Fenn [5]	Lee [6]	Fig. 3
basis	polynomial	polynomial	dual	AOP	AOP
AND	2	2	2	1	2
XOR	0	2	2	1	0
3XOR	1	0	0	0	1
Latch	7	7	7	3	5
number of cells	m^2	m^2	m^2	$(m+1)^2$	$m(m+1)/2$
latency	$3m$	$3m$	$3m$	$m+1$	$m/2+1$
critical path delay	D_A+D_{3X} $+D_L$	D_A+D_X $+D_L$	D_A+D_X $+D_L$	D_A+D_X $+D_L$	D_A+D_{3X} $+D_L$

Since our array needs fewer latches compared with that of [6], we find that the hardware complexity of the array in Fig. 3 is significantly reduced from the design in [6]. Also the latency is reduced from $m+1$ in [6] to $m/2+1$ in our case.

4 Linear Systolic Arrays for Cryptographic Purposes

The statements of Theorem 1, Theorem 2 and the corresponding circuits of basic cells imply that we may construct bidirectional linear systolic arrays with parallel-in parallel-out structures, which are quite different from other well known bit serial systolic arrays such as the design of Wang and Lin [3], or that of Yeh et al. [4]. Our linear systolic arrays are suitable for a cryptographic purpose because they have low latency, either $m+1$ or $m/2+1$, while the latency of other architectures [3,4,5] are $3m$. Though it was not noticed in [6], it is not difficult to see that we can construct a bidirectional linear systolic array with parallel-in and parallel-out structure which has a latency $m+1$ and a throughput rate $1/(m+1)$. The basic cell and the corresponding array are shown in Fig. 4.

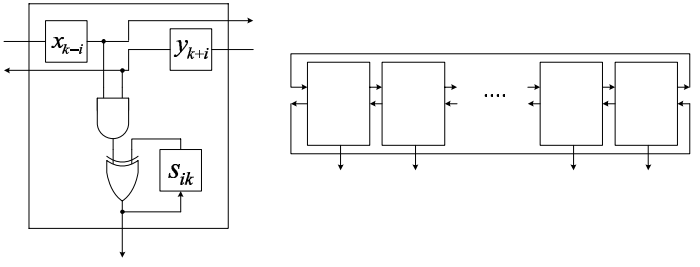


Fig. 4. Linear systolic array derived from Theorem 1

The basic cell in Fig. 4 shows the state of kth $(0 \le k \le m)$ cell after ith $(0 \le i \le m+1)$ clock cycle. The flip-flop for a partial summation has the value $s_{ik} = z_{2k} + \sum_{j=0}^{i-1} y_{k+j}x_{k-j}$. In particular, it is loaded with z_{2k} at the beginning, i.e. at the 0th clock cycle. Note that the initial values of x_{k-i} and y_{k+i} $(i = 0, 0 \le k \le m)$ are $x_0, x_1, \cdots, x_m$ and $y_0, y_1, \cdots, y_m$.

One may also use Theorem 2 to derive a linear systolic array with reduced latency $m/2+1$. The array is shown in Fig. 5. In Fig. 5, the state of kth $(0 \le k \le m)$ basic cell after ith $(0 \le i \le m/2+1)$ clock cycle is shown. In a similar way, notice that the value s_{ik} in the flip-flop for a partial summation has $s_{ik} = z_{2k} + \sum_{j=0}^{i-1}(y_{k+j}x_{k-j} + y_{k-j-1}x_{k+j+1})$ for $0 < i \le m/2$, and the final output is $s_{\frac{m}{2}k} + y_{k+m/2}x_{k-m/2}$ which is $(xy+z)_{2k}$. In this case, our array gives an output after $m/2+1$th clock cycle, which is 50 percent faster than the design in Fig. 4. One difference is that, in Fig. 5, we have a control signal to control the final summation of $y_{k+m/2}x_{k-m/2}$, which has the logic values of $m/2$ consecutive ONE

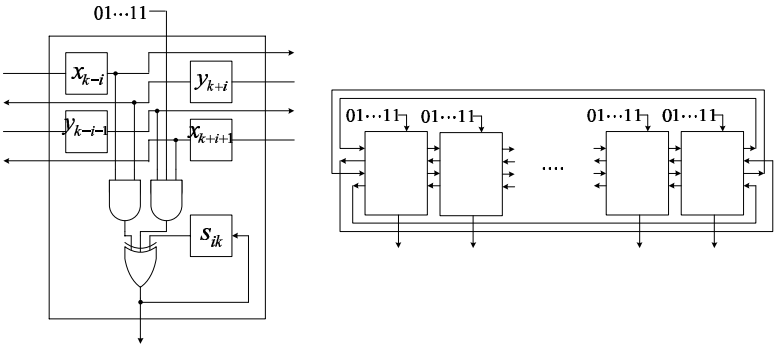


Fig. 5. Linear systolic array derived from Theorem 2

followed by ZERO. We compare our linear systolic arrays with other existing bit serial systolic arrays in Table 2. Note that D_M in Table 2 denotes the delay time of a 2-1 multiplexer.

Table 2. Comparison of our linear systolic arrays with other bit serial systolic multipliers.

	Wang [3]	Yeh [4]	Fenn [5]	Fig. 4	Fig. 5
basis	polynomial	polynomial	dual	AOP	AOP
AND	3	3	3	1	3
XOR	0	2	2	1	0
3XOR	1	0	0	0	1
MUX	2	2	3	0	0
flip-flop (Latch)	10	12	10	3	5
number of cells	m	m	m	$m+1$	$m+1$
latency	$3m$	$3m$	$3m$	$m+1$	$m/2+1$
critical path delay	$D_A+D_{3X}+D_L+D_M$	$D_A+D_X+D_L+D_M$	$D_A+D_X+D_L+D_M$	$D_A+D_X+D_L$	$2D_A+D_{3X}+D_L$
throughput rate	$1/m$	$1/m$	$1/m$	$1/(m+1)$	$1/(m/2+1)$

Compared with other linear systolic arrays in [3,4,5], the hardware complexity of our arrays are at least 50 percent reduced. Also the latency is 66 percent (resp. 83 percent) reduced in the case of Fig. 4 (resp. Fig. 5). Note that the area complexity of Fig. 5 is roughly twice of that of Fig. 4. However the latency and the throughput rate of Fig. 5 are twice better than those of Fig. 4.

5 Security of $GF(2^m)$ Determined by Irreducible AOP and the Density of Such m

To avoid possible known attacks such as Pohlig-Hellman method for discrete logarithm problem in a given finite field, one should be careful about the choice of suitable m for $GF(2^m)$. In general, it seems that the fields determined by irreducible AOP (equivalently, by optimal normal elements of type I) are not

actively used compared with the fields determined by optimal normal elements of type II or the Gauss periods of high order. One possible reason is that the degree m of a type one optimal normal element is even (composite). For elliptic curve cryptography, one should always choose m as a prime or an integer with at least one large prime factor to generate a point of large prime order on the given elliptic curve over the finite field.

Though there are not so many m for which an irreducible AOP exists and is applicable for reliable elliptic curve cryptographic protocols when compared with type II case, there still exist (and it seems that there are infinitely many of them) suitable m for our purpose. For example, we have the values ≥ 100 of $m = 106, 148, 172, 178, 226, 268, 292, 316, 346, 388, 466, 508, 556, 562, \cdots$ for which an irreducible AOP of degree m exists with a large prime factor dividing m, since $106 = 2 \cdot 53, 148 = 2^2 \cdot 37, 172 = 2^2 \cdot 43, 178 = 2 \cdot 89, 226 = 2 \cdot 113, \cdots$. Compare our list with the example of $m = 155 = 5 \cdot 31$ in elliptic curve specifications in *IEEE P1393*: specifications for PKC [9]. One more thing we have to consider is that we should choose m in such a way that $2^m - 1$ is not a product of small primes. This is necessary to avoid Pohlig-Hellman attack in $GF(2^m)$. By looking at the table of the factorization $2^m - 1$ in [10], we have a much better situation in this case since there are plenty of m for which a type I normal element exists and $2^m - 1$ has a sufficiently large prime factor.

Finally, it should be mentioned that the generalized 'Riemann Hypothesis' (See [14].) implies that there are infinitely many m for which an irreducible AOP of degree m exists. Let $a \neq 0, \pm 1$ be an integer which is not an rth power for any $r > 1$. Define $N_a(x)$ be the number of primes $p \leq x$ for which a is a primitive root $\pmod p$. In 1927, E. Artin conjectured that $N_a(x)$ is related to the following asymptotic formula,

$$N_a(x) \sim C(a)\frac{x}{\ln x},$$

where $C(a) = C_a C_{Artin}$ is a constant depending on a. That is, writing $a = a'b^2$ with a' square free, we have the constant C_a depending on a,

$$C_a = 1 \ if \ a' \not\equiv 1 \pmod 4, \quad C_a = 1 - \mu(a') \prod_{q|a'} \frac{1}{q^2 - q - 1} \ if \ a' \equiv 1 \pmod 4,$$

where $\mu(a')$ is the usual Möbius function and the product runs through all primes q dividing a'. The Artin constant C_{Artin} is expressed as

$$C_{Artin} = \prod_q (1 - \frac{1}{q^2 - q}) = 0.3739558 \cdots,$$

where the product runs through all primes. This conjecture was proved by Hooley [11] by using the generalized 'Riemann Hypothesis'. Later, a weaker form of Artin's conjecture was proved by Gupta and Murty [12] and by Heath-Brown [13] without using Riemann Hypothesis. However, at this moment, there is no known single example of a for which the conjecture of Artin is proved without

any extra assumption or hypothesis. Based on extensive computational evidence, it is generally believed that Riemann Hypothesis and also Artin's conjecture are true. Therefore, to apply the conjecture to our case, let $a = 2$. Then we have $C_a = 1$ and thus $C(a) = C_{Artin} = 0.3739558\cdots$. Consequently, by using the well known 'Prime Number Theorem' [14] saying

$$\lim_{x\to\infty} \frac{\pi(x)}{x/\ln x} = 1,$$

where $\pi(x)$ is the number of primes $\leq x$, we conclude that 2 is a primitive root (mod p) for approximately $37.39558\cdots$ percent of all primes p. And for those primes p, $m = p - 1$ gives the values of m for which a type I optimal normal element (equivalently, an irreducible AOP) of degree m exists.

6 Conclusions

In this paper, we proposed a low complexity and a low latency systolic arrays using an irreducible all one polynomial (AOP) in $GF(2^m)$. We showed that the proposed bit parallel array has a considerable advantage in terms of latency and hardware complexity when compared with the design in [6]. The latency of our bit parallel array in Fig. 3 is $m/2+1$ while the latency in [6] is $m+1$. Moreover by comparing the gate areas, we find that the hardware complexity of Fig. 3 is significantly reduced from the design in [6]. Also, we presented new linear systolic arrays, Fig. 4 and 5, using an irreducible AOP, which are applicable for a cryptographic purpose where a large m for $GF(2^m)$ is used. Our linear systolic arrays have significantly reduced latency and hardware complexity compared with other existing linear systolic arrays as shown in Table 2. Since the design in Fig. 5 has a twice high throughput rate $1/(m/2+1)$ with at least 50 percent reduced hardware complexity compared with those of [3,4,5], it can be used in many hand-held devices for time critical applications.

Acknowledgement. This paper was supported by Faculty Research Fund, Sungkyunkwan University, 2002.

References

1. E.R. Berlekamp, "Bit-serial Reed-Solomon encoders," *IEEE Trans. Inform. Theory*, vol. 28, pp. 869–874, 1982.
2. T. Itoh, and S. Tsujii, "Structure of parallel multipliers for a class of finite fields $GF(2^m)$," *Information and computation*, vol. 83, pp. 21–40, 1989.
3. C.L. Wang and J.L. Lin, "Systolic array implementation of multipliers for finite fields $GF(2^m)$," *IEEE Trans. Circuits Syst.*, vol. 38, pp. 796–800, 1991.
4. C.S. Yeh, I.S. Reed, and T.K. Troung, "Systolic multipliers for finite fields $GF(2^m)$," *IEEE Trans. Computers*, vol. C-33, pp. 357–360, 1984.
5. S.T.J. Fenn, M. Benaissa, and D. Taylor, "Dual basis systolic multipliers for $GF(2^m)$," *IEE Proc. Comput. Digit. Tech.*, vol. 144, pp. 43–46, 1997.

6. C.Y. Lee, E.H. Lu, and J.Y. Lee, "Bit parallel systolic multipliers for $GF(2^m)$ fields defined by all one and equally spaced polynomials," *IEEE Trans. Computers*, vol. 50, pp. 385–393, 2001.
7. A. Reyhani-Masoleh and M.A. Hasan, "A new construction of Massey-Omura parallel multiplier over $GF(2^m)$," *IEEE Trans. Computers*, vol. 51, pp. 511–520, 2002.
8. S.T.J Fenn, M.G. Parker, M. Benaissa, and D. Taylor, "Bit-serial multiplication in $GF(2^m)$ using irreducible all one polynomials," *IEE Proc. Comput. Digit. Tech.*, vol. 144, pp. 391–393, 1997.
9. IEEE P1363: Standard specifications for public key cryptography, 1999.
10. J. Brillhart, D.H. Lehmer, J.L. Selfridge, B. Tuckerman, and S.S. Wagstaff Jr., "Factorizations of $b^n \pm 1$, $b = 2, 3, 5, 7, 10, 11, 12$ up to High Powers," *Contemporary Mathematics*, vol. 22, American Mathematical Society, 1988.
11. C. Hooley, "On Artin's conjecture," *J. reine angew. Math.*, vol. 225, pp. 209-220, 1967.
12. R. Gupta and M. Ram Murty, "A remark on Artin's conjecture," *Inventiones Math.*, vol. 78, pp. 127–130, 1984.
13. D. Heath-Brown, "Artin's conjecture for primitive roots," *Quart. J. Math.*, vol. 37, pp. 27-38, 1986.
14. G. Tenenbaum and M.M. France, "The Prime Numbers and Their Distribution," *translated by P.G. Spain*, Ameriacn Mathematical Society, 2000.
15. A.J. Menezes, I.F. Blake, S. Gao, R.C. Mullin, S.A. Vanstone, and T. Yaghoobian, "Applications of Finite Fields," Kluwer Academic Publisher, 1993.

Price Driven Market Mechanism for Computational Grid Resource Allocation

Chunlin Li, Zhengding Lu, and Layuan Li

Department of Computer Science, Wuhan University of Technology, Wuhan 430063, P.R. China Department of Computer Science, Huazhong University Of Science &Technology, Wuhan 430074, P.R.China

chunlin74@tom.com, jwtu@public.wh.hb.cn

Abstract. This paper presents a price driven market mechanism for resource allocation in computational grid. A system model is described that allows agents representing various grid resources, which owned by different real world enterprises, to coordinate their resource allocation decisions without assuming a priori cooperation. The grid task agents buy resources to complete tasks. Grid resource agents charge the task agents for the amount of resource capacity allocated. Given grid resource agent's pricing policy, the task agent optimization problem is to complete its job as quickly as possible when spending the least possible amount of money. This paper provides a price-directed proportional resource allocation algorithm for solving the grid task agent resource allocation problem. Experiments are made to compare the performance of the price-directed resource allocation with conventional Round-Robin allocation.

1 Introduction

Grid Computing is an emerging technology that promises to unify resources and computing power in many organizations together. It is widely used to solve large-scale problems in engineering and science area [1]. One important problem in such environments is the efficient allocation of computational resources [2]. Markets have emerged as a new paradigm for managing and allocating resources in complex systems. Markets are appropriate for decentralized systems because once a currency exchange protocol is established, negotiations can occur simultaneously at various nodes without the necessity of a central authority [3~7]. Scalability is another advantage as new resources and new resource users can be added simply by establishing the ability to receive or give currency. Also, prices serve as useful low-dimensional feedback for control. Market-based control has been applied to factory scheduling, manufacturing systems, energy distribution and pollution management [8~9]. Agent-based technique that is becoming increasingly popular as a means of tackling distributed resource allocation tasks is market-based control [10]. In such systems, the producers and consumers of the resources of a distributed system are modeled as the self-interested decision-makers described in standard microeconomic theory [11]. The individual agents in such an economic model decide upon their demand and supply of resources, and on the basis of this the market is supposed to generate an equilibrium distribution of resources that maximizes social welfare.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 117–126, 2004.
© Springer-Verlag Berlin Heidelberg 2004

In this paper, a market-based approach to computational grid resource allocation is presented. The grid task agents buy resources to complete tasks. Grid resource agents charge the task agents for the amount of resource capacity allocated. Given grid resource agent's pricing policy, the task agent optimization problem is complete its job as quickly as possible when spending the least possible amount of money. Then, price-directed market-based algorithm for solving the grid task agent resource allocation problem is provided. Experiments are made to compare the performance of the price-directed resource allocation with conventional Round-Robin allocation.

2 System Model

The overall system model consists of three layers. The lower layer is the underlying grid resource. Resources on this layer are owned and allocated by grid resource agents deployed at the nodes in the grid. The top layer is the system' s interface to grid user. The middle layer is the agent-based grid resource management system. It consists of three types of agent and market institution that allocates resources in response to the selling of grid resource agent and buying behavior of the grid task agents. The third layer is the user layer at which grid request agents provide interfaces to the grid user' request. Grid resource agents sell the underlying resources of the grid. A task agent that represents the grid user makes buying decisions within budget constraints to acquire computation resources.

The system model makes use of two economic agent types: (1) the grid resource agents that represent the economic interests of the underlying resources of the computational grid, (2) the grid task agents that represent the interests of grid user using the grid to achieve goals. A grid resource agent is used at the source node in the grid and is deployed at the entry node. The Grid resource agents have varied computational resource capacity, and the computational resource capacity is shared among the grid task agents. The grid resource agents charge the task agents for the portion of the computational resource capacity occupies. We assume that the grid resource agents of a grid does not cooperate, probably due to high messaging and processing overheads associated with cooperative allocating. Instead, they act non-cooperatively with the objective of maximizing their individual profits. The grid resource agents compete among each other to serve the task agents. The task agents do not collaborate either, and try to purchase as much computational resource as possible with the objective of maximizing their net benefit. The agents communicate by means of a simple set of signals that encapsulate offers, bids, commitments, and payments for resources. We couple the resources and payments with the offers and requests respectively. This reduces the number of steps involved in a transaction (committing agents to their payments and offers ahead of the market outcome), and so increases the speed of the system's decision making. To enforce these rules the interactions between the two agent types are mediated by means of market mechanisms. In our market mechanisms, agent communication is restricted to setting a price on a single unit of a known grid resource. Therefore, agents set their prices solely on the basis of their implicit perception of supply and demand of grid resource at a given time. When a resource is scarce, grid task agents have to increase the prices they are willing to buy, just as resource agents decrease the price at which they are

willing to offer the resource. In our model, agents perceive supply and demand in the market through price-directed market-based algorithm that will be described in Section 4.

3 Grid Task Agent's Optimal Strategy

The grid task agents buy resources to complete tasks. Grid resource agents charge the task agents for the amount of resource capacity allocated. However, there are multiple grid task agents competing to buy the grid resource agent's computation resource. We investigate the effect of this competition on the system model. Specifically, we show that such a price competition leads to the optimal grid resource allocation strategy for the grid task agents. This approach provides a dynamical and distributed algorithm for determining the resource allocation in the grid that will be presented in section 4. In this section we find the task agent's optimal allocating strategy under a certain grid resource agents pricing scheme.

Let u_i^j be the price of the *i*th task agent paid to *j*th resource agent. Let u_i be the total investment of the *i*th task agent, which is defined in (3.1). Let p_j denote the price of the unit computational resource in resource agent *j*. Let the pricing policy, p =($p1$,$p2$,... , p_j), denote the set of unit computational resource prices of all the resource agents in the grid.

$$u_i = \sum_j u_i^j \tag{3.1}$$

Let x_i^j be the resource units allocated to task agent *i* by resource agent *j*. If *i* th task agent' s payment in the *j*th resource agent is u_i^j , then the total computation resource units allocated to task agent *i* is

$$x_i^j = \frac{u_i^j}{p_j}. \tag{3.2}$$

The goal of each task agent is to complete its job as quickly as possible when spending the least possible amounts of money. q_{ij} is the size of *i*-th task agent' s *j*-th job. c_j is the capacity in computational units of *j*-th grid resource agent. Since the grid user wishes to minimize both the time, $\sum_{J=1}^{N} \frac{q_{ij}}{c_j x_i^j} + D$, and money $\sum_j u_i^j$ it spends. The utility function $U(x_i^j)$ of the grid task agent is defined as (3.3).

$$U(x_i^j) = -K\left(\sum_{J=1}^{N} \frac{q_{ij}}{c_j x_i^j} + D\right) - \sum_j u_i^j. \tag{3.3}$$

Where *D* is the delay, which includes waiting times, transfer times between various nodes in the grid. *K* is the relative importance of costs and times to complete grid task, an agent with larger value of *K* would indicate a greater preference to reduce its completion time. When *K* is 1, meaning that costs and times are equally important.

Every grid task agent tries to maximize itself benefit regardless of others subject to the availability of budgets and complete time limits. For a given grid resource pricing policy *P*, the task agent optimization problem (S) can be written as (3.4).

$$(S)\ \begin{matrix} MaxU(x_i^j) \\ \text{s.t. } E_i \geq \sum_j x_i^j p_j \end{matrix} \tag{3.4}$$

Constraint is a budget constraint, which says that the aggregate sum of all costs of each task agent cannot exceed its total budget. *Ei* is endowment given to an agent. Our objective is to choose optimal x_i^j .

$$\sum_{i=1}^{N} x_i^j = 1 \tag{3.5}$$

(3.5) Indicates a grid resource is divisible, that can be shared among many grid task agents. We substitute $x_i^j = \frac{u_i^j}{p_j}$ into $U(x_i^j)$ to obtain (3.6)

$$U(x_i^j) = -K(\sum_{J=1}^{N} \frac{q_{ij}}{c_j x_i^j} + D) - \sum_j x_i^j p_j \tag{3.6}$$

We compute the optimum by deriving the derivative of $U(x_i^j)$ with respect to x_i^j as (3.7).

$$U'(x_i^j) = \frac{dU(x_i^j)}{d x_i^j} = K \sum_{J=1}^{N} \frac{q_{ij}}{c_j (x_i^j)^2} - p_j \tag{3.7}$$

Then, the second derivative of $U(x_i^j)$ with respect to x_i^j is (3.8).

$$U''(x_i^j) = \frac{d^2 U(x_i^j)}{d (x_i^j)^2} = -K \sum_{J=1}^{N} \frac{q_{ij}}{c_j (x_i^j)^3} \tag{3.8}$$

$U''(x_i^j) < 0$ is negative due to $0 < x_i^j < 1$ .The extreme point is the unique value maximizing the agent's utility and is optimal resource demand for grid resource agent. Grid task agent' s utility is a convex function of x_i^j . A common method of optimizing convex function is to apply Lagrangian. The Lagrangian for the task agent' s utility is *L*(x) (3.9).

$$L(x_i^j) = -K(\sum_{J=1}^{N} \frac{q_{ij}}{c_j x_i^j} + D) - \sum_j x_i^j p_j - \lambda(\sum_j x_i^j p_j) \tag{3.9}$$

Where λ is the Lagrangian constant. From Karush-Kuhn-Tucker Theorem we know that the optimal solution is given $\partial L(x) / \partial x = 0$ for $\lambda > 0$.

$$\partial L(x_i^j) / \partial x_i^j = K \sum_{J=1}^{N} \frac{q_{ij}}{c_j (x_i^j)^2} - (1+\lambda) p_j \tag{3.10}$$

Let $\partial L(x_i^j) \Big/ \partial x_i^j = 0$ to obtain $x_i^j = \left(\frac{Kq_{ij}}{(1+\lambda) p_j c_j}\right)^{1/2}$ (3.11)

Using this result in the constraint equation, we can determine $\theta = \lambda + 1$ as

$$(\theta)^{-1/2} = \frac{E_i}{\sum_{k=1}^{N} p_k \left(\frac{Kq_{ik}}{c_k p_k}\right)^{1/2}} \quad (3.12)$$

We substitute (3.12) into (3.11) to obtain $x_i^{j*} = \frac{\left(\frac{q_{ij}}{p_j c_j}\right)^{1/2} E_i}{\sum_{k=1}^{N} p_k \left(\frac{q_{ik}}{c_k p_k}\right)^{1/2}}$ (3.13)

x_i^{j*} is the unique optimal solution to the optimization problem (S).

4 Price-Directed Grid Resource Allocation Algorithms

We design a price-directed market-based algorithm for solving the grid task agent resource allocation problem. In this algorithm, an initial set of prices is announced to the task agent. The task agents determine their resource demands according to these prices. The task agents request these resources capacity from the resource agents. Prices are then iteratively changed to accommodate the demands for resources until the total demand equals to the total amount of resources available.

The detail of whole process can be described as follows: Grid resource agents announce a set of initial prices $P = (p_1, p_2, \ldots\ldots p_j)$, each grid task agent *i* calculates its optimal resource demand for grid resource agent. Then, forward these resource demands to the grid resource agents. At iteration *n*, each grid resource agent *j* updates its price according to the grid task agent's demands.

$$p_j^{(n+1)} = \max\{\varepsilon, p_j^{(n)} + \eta(x^j P^{(n)} - C_j)\}$$

Where $x^j = \sum_i x_i^j$, *n* is the step size. Let $\varepsilon > 0$ be a sufficiently small constant preventing prices to approach zero. Thus, if the total demand $\sum_i x_i^j$ is greater than the cache capacity C_j, then the new price $p_j^{(n+1)}$ is increased, otherwise it is decreased. Grid resource agent announces the new prices $P^{(n)}$ to the grid task agents. This cycle stops until the total demand equals to the total amount of resources available, $P^{(n)}$ are the set of prices at the equilibrium.

Algorithm 1: Price-directed resource allocation algorithm

Grid task agent part algorithm {

If a task submitted {

For every task agent participating in competing resources {

send request to grid resource agent; }

}

If grid resource agent reply comes in {

Store the reply;

If all price replies for this task are received {

For all price replies

Repeat { n= n+1;

calculates its optimal resource demand;

Send resource demands to the grid resource agents;}

Until the total demand equals to the total amount of resources available. }

Send payment to resource grid agent;

Get allocated resource.

}

Grid resource agent part algorithm {

Grid resource agents announce a set of initial prices: $P = (p_1, p_2, \ldots\ldots p_j)$

if grid task agent reply comes in {

For each grid resource agent {

n=n+1;

$$p_j^{(n+1)} = \max\{\varepsilon, p_j^{(n)} + \eta(x^j P^{(n)} - C_j)\}$$

$$Cj = \sum_j x_i^j$$

Announce the new prices $P^{(n)}$ to the grid task agents;

} Until the total demand equals to the total amount of resources available.

}

5 Experiments

The goal of this experiment is to compare the performance of a decentralized economic approach based on the price-directed resource allocation algorithm with conventional Round-Robin allocation algorithm. To do this, both approaches are evaluated experimentally by means of simulations. In the Round-Robin allocation scheme, no pricing is used. The incoming task queries are matched with the next available resource offer, which meets the task's constraints but which is usually not the best. First, we introduce the configuration of simulation, then, give the experiment design and results.

The simulator was developed to test the price-directed allocation algorithm. It is implemented on top of the JAVASIM network simulator. Different agent types can be instantiated, namely grid client, grid task agents, and grid resource agents. Grid resources to be allocated encompass computation service access, bandwidth and storage. The experiment is to study characteristics of price-directed allocation algorithm with Round-Robin algorithm in terms of response time and resource allocation efficiency. Grid systems are randomized in various sizes: 100, 500, 1000, and 2000 nodes. In the experiments we change some of test parameters, such as the size of grid that is denoted by *S* in bellow figures, resource' requests intensity is denoted by *I*. The experiment is to randomly submit 250 grid requests and schedule them to the specific grid resource based on price-directed resource allocation and Round-Robin allocation. Arrival time of each resource request is determined randomly in exponential distribution with mean of 200ms, but we will change the values of arrival time when testing effect of requests intensity on response time and resource allocation efficiency. All nodes are initially no loads. During the time of experiment, grid resource requests are generated by the grid user agent. After this initial period, the number of tasks that is statistically expected to be generated during an interval of 100 time units is considered in the result. There are 25 grid resource agents in the system. All gird resource agents have the same resource size denoted by *R*, set *R*=100. Each measurement is run 30 times with different seeds. These experimental configurations are to bring up performance of resource allocation algorithm as many as possible. Interesting variables are recorded and plot average results in Fig.1, Fig.2 for response times and resource allocation efficiency respectively.

Firstly, we have measured the response times of price-directed allocation and Round-Robin allocation when using the following parameters for the test: (I=200ms). Response time measures the time observed by the grid client to access the requested grid resources. It is influenced by the size of the grid, the available connections and bandwidth, and especially by the necessary mechanisms to establish a working link between grid task agent and grid resource agent. From the results in Fig. 1, for Round-Robin allocation, the response time value seems to depend on the size of grid. Price-directed allocation and Round-Robin allocation present the good results for this small size grid. But, when the size of grid is larger, Round-Robin allocation is decreasing quickly; the response time using price-directed allocation can be as much as 44% shorter than that using the Round-Robin allocation. On big grid, Round-Robin allocation takes more time to allocate appropriate resources. As shown in Figure 1, for different size grid, the price-directed allocation outperforms the conventional Round-Robin allocation.

Secondly, we measured the resource allocation efficiency of price-directed allocation and Round-Robin allocation when using the following parameters for the test: (I=200ms). Resource allocation efficiency indicates the ratio of grid resource requests, for which the grid resource agent grants to provide a resource, to all sent grid resource requests. In other words, it measures how many requests a grid client has to send until a resource agent accepts its demand and grants access. As the request messages waste up bandwidth, higher resource allocation efficiency is deemed to be better both for the individual grid client agent and for the whole grid as a whole. The

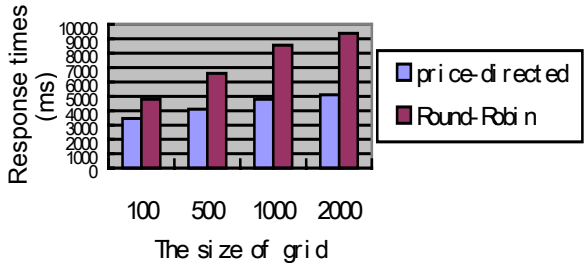


Fig. 1. Comparison of response time

results are shown in Fig.2. It becomes clear that both allocation schemes work best under small size grid. The Round-Robin allocation achieves to match nearly 98% of all requests in small size grid scenario, with price-directed allocation closely behind. However, as grid size increases, the Round-Robin allocation soon loses comparably more performance than the price-directed allocation. Under large size grid, the decrease of the results for Round-Robin allocation is lower than in the small size. Resource allocation efficiency using price-directed allocation is as much as 27% larger than that using the Round-Robin allocation. Varying grid size, result decreases for both methods similarly.

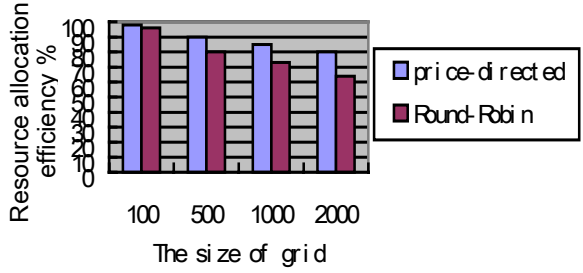


Fig. 2. Resource Allocation Efficiency

From above performance comparisons, we can get some conclusions. In most of the test cases, the price-directed allocation is more efficient than the Round-Robin allocation to allocate grid resource in test application. When grid size is creasing, it is have more merits to use the price-directed allocation to schedule grid resource; the price-directed allocation has better performance than usual Round-Robin allocation.

6 Conclusions

This paper presents a market-based approach to computational grid resource management. A realistic model for the relationship between the grid task agent and grid resource agent is presented. The grid task agents buy resources to complete tasks.

Grid resource agents charge the task agents for the amount of resource capacity allocated. However, there are multiple grid task agents competing to buy the grid resource agent's computation resource. Given grid resource agent's pricing policy, the task agent optimization problem is provided. This paper provides a price-directed market-based algorithm for solving the grid task agent resource allocation problem. The results of experiment show the price-directed allocation has better performance than usual Round-Robin allocation.

References

[1] I. Foster and C. Kesselman, The Grid : Blueprint for a New Computing Infrastructure, Morgan Kaufmann, 1999.

[2] K. Krauter, R. Buyya, and M. Maheswaran, A Taxonomy and Survey of Grid Resource Management Systems, Software: Practice and Experience, February 2002, Vol32 (2), 135-164

[3] S. Lalis and A. Karipidis, An Open Market-Based Framework for Distributed Computing over the Internet, Proceedings of the First IEEE/ACM International Workshop on Grid Computing (GRID 2000), Dec. 17, 2000, Bangalore, India, Springer Verlag Press, 2000, 36-46.

[4] R. Wolski, J. Plank, J. Brevik, and T. Bryan, Analyzing Market-based Resource Allocation Strategies for the Computational Grid, International Journal of High-performance Computing Applications, Sage Publications, 2001,Vol 15(3), 258-281

[5] Jonathan Bredin, David Kotz, and Daniela Rus. Utility driven mobile-agent scheduling. Technical Report PCS-TR98-331, Dartmouth College, 1998.

[6] Li Chunlin, Li Layuan, An Agent-oriented and Service-oriented Environment for Deploying Dynamic Distributed Systems, Journal Computer Standard and Interface, Elsevier, Vol 24/4, pp. 321-334, Sept, 2002

[7] Donald F. Ferguson, Yechiam Yemini, Christos Nikolaou: Microeconomic Algorithms for Load Balancing in Distributed Computer Systems. Proceedings of the 8th International Conference on Distributed Computing Systems, San Jose, IEEE-CS Press, 1988, 491-499

[8] Mark Carman, Floriano Zini, Luciano Serafini et al, Towards an Economy-Based Optimization of File Access and Replication on a Data Grid, CCGrid 2002, Berlin, IEEE Computer Society Press, 2002, 340-345

[9] R. Buyya, H. Stockinger, J. Giddy, and D. Abramson, Economic Models for Management of Resources in Peer-to-Peer and Grid Computing, In Proceedings of International Conference on Commercial Applications for High-Performance Computing, SPIE Press, 2001, 13-25.

[10] S. H. Clearwater, Market-Based Control A Paradigm for Distributed Resource Allocation, Ed. Clearwater S. H., World Scientific Press. 1996

[11] H.R. Varian, Microeconomic Analysis (Third Ed.), W.W. Norton & Company Inc. (1992)

[12] Li Chunlin, Zhengding Lu, Li Layuan, Apply Market Mechanism to Agent-Based Grid Resource Management, International Journal of Software Engineering & Knowledge Engineering, World Scientific Publishing, Vol. 13/ 3, pp. 327-340, June, 2003

[13] Li Chunlin, Lu Zhengding, Li Layuan , Zhang Shuzhi, A Mobile Agent Platform Based On Tuple Space Coordination, Journal of advances in engineering software, Elsevier, 2002, Vol 33(4), 215-225.

[14] Li Chunlin, Li Layuan, Integrate Software Agents And CORBA In Computational Grid, Journal of Computer Standards and Interfaces, Elsevier, Vol 25/4, pp. 357-371, August, 2003
[15] Li Chunlin, Lu zhengding, Li layuan. Design and Implementation of a Distributed Computing Environment Model for Object_Oriented Networks Programming, Journal of Computer Communications, Elsevier, Vol 25/5, pp 517-522, Mar 2002
[16] Li Chunlin, Li Layuan, Agent Framework to Support Computational Grid, Journal of System and Software, Elsevier, Vol 70/1-2 pp. 177-187, February, 2004

A Novel LMS Method for Real-Time Network Traffic Prediction

Yang Xinyu, Zeng Ming, Zhao Rui, and Shi Yi

Dept. of Computer Science and Technology, Xi'an Jiaotong University,
710049 Xi'an, P.R.C
yxyphd@mail.xjtu.edu.cn

Abstract. Real-time traffic prediction could give important information to both network efficiency and QoS guarantees. On the basis of LMS algorithm, this paper presents an improved LMS predictor – EaLMS (Error-adjusted LMS) – for fundamental traffic prediction. The main idea of EaLMS is using previous prediction errors to adjust the LMS prediction value, so that the prediction delay could be decreased. The prediction experiment based on real traffic trace has proved that for short-term traffic prediction, compared with traditional LMS predictor, EaLMS significantly reduces prediction delay, especially at traffic burst moments, and avoids the problem of augmenting prediction error at the same time.

1 Introduction

Traffic prediction is an important research field of the traffic engineering. Recent work in this area mainly includes using time series analysis model [1], artificial neural-network method [2-3], wavelet method [4], etc. Most of the above methods need history traffic record, and have large complexity of calculation. For short-term real-time prediction, efficient adaptive methods are needed. Among them, least-mean-square (LMS) algorithm is of particular interest [5,6,7and 8] due to its simplicity and relatively good performance.

One problem associated with LMS is its compromise between convergence speed and tracking performance. While applying LMS to traffic prediction, the problem exists between prediction delay and prediction error. On the one hand, a larger step size will reduce prediction delay, but bring the problem of convergence that leads to increasing prediction error; on the other hand, a smaller step size gives less prediction error but a longer prediction delay.

The fundamental traffic, obtained by smoothing filtering, preserves the main characteristic of original traffic, and is relatively more stable and more suitable for applying LMS predictor. The authors' work attempts to improve the LMS predictor for fundamental traffic -- by using previous prediction error to adjust LMS prediction value – which is called Error-adjusted LMS in this paper. Experiment based on fundamental traffic of real network trace has proved that for short-term real-time prediction, compared with traditional LMS predictor, EaLMS significantly reduces prediction delay, and avoids the problem of augmenting prediction error at the same time.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 127–136, 2004.
© Springer-Verlag Berlin Heidelberg 2004

The paper is organized as follows. Section 2 briefly introduces LMS algorithm and some improvements. Section 3 describes EaLMS predictor, and Section 4 is an analysis of prediction experiment. Section 5 discusses the influence of Low-pass filter to traffic prediction, and Section 6 contains a short conclusion.

2 LMS and the Idea of EaLMS

LMS is one of the most popular algorithms in adaptive signal processing, which was proposed by WIDROW and HOFF. The algorithm is of the form,

$$\hat{\mathbf{W}}(n+1) = \hat{\mathbf{W}}(n) + \frac{1}{2}\mu[-\hat{\nabla}(n)] = \hat{\mathbf{W}}(n) + \mu e(n)\mathbf{X}(n) \ . \tag{1}$$

If applied with adaptive AR(p) model, the LMS predictor is on form,

$$e(t) = x(t) - \varphi^t(t)\mathbf{x}(t-1) \ . \tag{2}$$

$$\varphi(t+1) = \varphi(t) + \mu\mathbf{x}(t-1)e(t) \ . \tag{3}$$

$$\varphi(t) = [\varphi_1, \varphi_2, ..., \varphi_p]^t$$

where $\mathbf{x}(t-1) = [x(t-1), x(t-2), ..., x(t-p)]^t$

Here μ is the step size. In standard LMS, μ is a constant and its value determines the speed of adaptive process. The condition of convergence is 0<μ< max, and max is the max eigenvalue of correlation matrix R.

The learning speed of LMS method is controlled by step size: the larger μ is, the faster the convergence speed is. However, an excessive μ can affect the convergence of algorithm and will augment steady state misjudgment. In LMS predictor, the tradeoff between prediction delay and prediction error is also controlled by step size. On the one hand, a larger step size will reduce prediction delay, but bring problem of convergence that leads to increasing prediction error; on the other hand, a smaller step size gives less prediction error but a longer prediction delay. To solve the conflict between learning speed and steady state misjudgment, many improved LMS algorithms are proposed. VSS-LMS [9], proposed by Kwong and Johston, is a typical one. VSS-LMS uses a variable step size to reduce the tradeoff between misadjustment and tracking ability of the fixed step size LMS.

For real-time traffic prediction, a quick response speed is very needed, especially at traffic burst moments. The fundamental traffic obtained by smoothing filtering, preserves the main characteristic of original traffic, and is relatively more stable and more suitable for applying LMS predictor. The authors' work attempts to improve the LMS predictor for fundamental traffic – by using previous prediction error to directly adjust LMS prediction value – which is called Error-adjusted LMS in this paper. EaLMS predictor does not make any modification to LMS algorithm, but only adds an adjustment quantity to LMS prediction result, and this adjustment is a function of previous prediction error. Calculation of the adjustment quantity uses some statistics

parameters of prediction errors and these parameters can be estimated through computing process.

While using adaptive AR model for prediction, LMS uses prediction error e(t) to modify φ, VSS-LMS uses e(t) to change φ and step size μ, and EaLMS uses e(t) to adjust φ and prediction value. Tab.1 shows the difference of these three methods on this point.

Table 1. Adjusted objects in three LMS prediction methods

predictors	LMS	VSS-LMS	EaLMS
e(t) direct object	e→φ	e→μ	e→φ
		e→φ	e→ $\hat{x}(t+1)$
e(t) indirect object	φ→ $\hat{x}(t+1)$	μ→φ→ $\hat{x}(t+1)$	φ→ $\hat{x}(t+1)$
		φ→ $\hat{x}(t+1)$	

3 EaLMS Predictor for Network Traffic

The objective of EaLMS predictor is mainly to predict network fundamental traffic. Measured traffic series usually have strong bursts and fluctuations. Fundamental traffic, obtained by passing low-pass smoothing filter, preserves the main characteristic of original traffic, and removes most of short-term random variations.

According to the presentation above, EaLMS predictor adds an adjustment quantity ε(t) to the LMS prediction result $\hat{x}(t+1)$. The key problem of EaLMS predictor is to compute ε(t). It is necessary to find a proper way to figure out adjustment quantity, which can expedite tracking speed, but will not produce strong fluctuation on prediction. Prediction delay at burst moments best reflects tracking performance of predictor, and prediction error can be examined by the RMSE (root mean square error) between real traffic and prediction value. Another requirement for EaLMS predictor is using adaptive parameters to different signals, instead of designating parameters artificially. A brief summary of the above discussion, gives two criteria of EaLMS predictor performance:

- Decrease of prediction delay (especially at burst moments);
- Reduction of prediction error (at least, not augmentation).
  One general applicability condition:
- Less designated parameters as possible.

3.1 Traffic Trace and Preprocessing

Throughout this paper, we use the TCP traffic data of [10]. The two traces used are LBL-TCP-3 and LBL-PKT-4. These traces respectively contain an hour's and two

hours' wide-area TCP traffic between the Lawrence Berkeley Laboratory and the rest of the world. The network trace is converted to a time series of traffic per second (bytes/s) at 1-second time scale. The original traffic is passed though smoothing filter (e.g., 10-degree) to get fundamental traffic, served as the signal to predict, noted as z(t).

To improve the performance, before applying LMS algorithm, normalization is employed to series z(t), i.e. $x(t) = (z(t) - \mu) / \sigma$, here mean μ and standard deviation σ are need to be estimated. Then LMS predictor is executed on x(t), and through corresponding conversion we get the prediction value of z(t). For the sake of real-time prediction, we used a dynamic method on parameter estimation, i.e. at each moment t, with the reception of z(t), adjustment is made to μ(t) and σ(t) as

$$\mu(t) = \frac{1}{t}\sum_{i=1}^{t} z(i) = \frac{1}{t}((t-1)\cdot\mu(t-1) + z(t)) \tag{4}$$

$$\sigma^2 = \mathrm{var}(t) = \frac{1}{t}\sum_{i=1}^{t}(z(i)-\mu(t))^2 \approx \frac{1}{t}((t-1)\cdot\mathrm{var}(t-1) + (z(t)-\mu(t))^2) \tag{5}$$

Estimation of other statistical parameters in EaLMS, such as σe(t) and conditional probability of sign continuity in the following section 3.3, uses the similar method.

3.2 Analysis of e(t) in LMS Predictor

This experiment uses the fundamental traffic Lbl-pkt-4 at 1-second time scale, obtained by passing 10-degree smoothing filter. The degree of AR model is 3, and μ = 0.01. One-step prediction value is denoted as za1(t), prediction interval (1, 3600), and e(t) denotes the prediction error x(t) of LMS predictor.

As shown in Fig.1, absolute value of e(t) corresponds to prediction error of LMS predictor. If at several continuous moments, e(t) has a same positive or negative sign, it will probably indicate a persistence of traffic variation trend (augment of descent). At this moment, LMS predictor usually has an obvious delay compared with real traffic (in Fig.1(a), 1200~1210s, 1220~1230s and 1230s~1240s). This phenomenon is partially due to the low-pass filter.

Statistical calculation on e(t) through (1, 3600) shows, μe=-0.0123,σe=0.2180. Fig.2 illustrates the distribution of e(t), and the probability distribution of P{|e(t)|/σe<n}. According to the statistics of large data, we can assume that e(t) approximately obey a Normal distribution N(0, σe) (except the initial period of LMS startup). Based on above analysis, we attempt to calculate adjustment quantity by absolute value and sign of e(t).

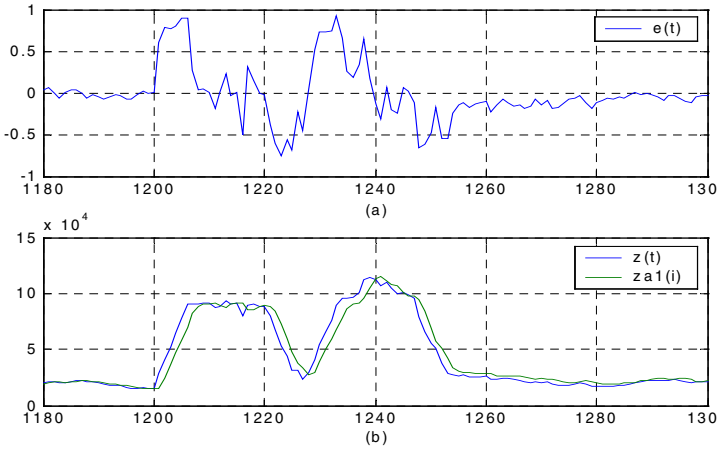


Fig. 1. Error *e(t)* of LMS (a) and one-step prediction value *za1(t)* (b)

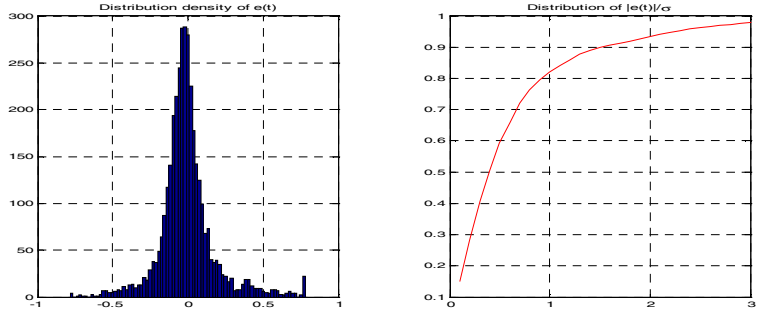


Fig. 2. Distribution density of *e(t)* (left) and distribution probability of $|e(t)|/\sigma_e$ (right)

3.3 Calculating Adjustment ε(t) by e(t)

The main idea of error-adjustment is to estimate the traffic variation trend according to the sign continuity and absolute value of e(t). Adjustment quantity ε(t) is added to the LMS prediction value, so that the new predictor could follow the variation of traffic more quickly, or even forecast it in advance. The adjustment quantity is determined by two elements -- absolute value of e(t) and its sign continuity, i.e. the product of two factors -- sign(t) and value(t), as

$$\varepsilon(t) = sign(t) * value(t) * e(t) \tag{6}$$

Sign continuity – factor sign(t). Factor sign(t) is decided by the sign continuity of prediction error. We examine statistical characteristics of e(t) on its sign continuity. Define n(t) as the count of e(t) which has the continuously same sign at t moment. n(t) = 1, means e(t) and e(t-1) have different sign; n(t)=2, e(t) and e(t-1) have same sign; n(t)=3, e(t), e(t-1) and e(t-2) have same sign. n(t) signifies the duration of period that e(t) keeps the same sign, n(t) 1. If we use random variable N to denote the value of n(t), statistic calculation on prediction error in section 3.2 gives Table 2.

Table 2. Probability of sign continuity of e(t)

Probability	conditional probability	*sign(t)*
P{N=1}=26.75%	--	1.0
P{N>=2}=73.14%	--	2*0.73
P{N>=3}=55.25%	P{N>=3|N>=2}=75%	2*0.75
P{N>=4}=43.58%	P{N>=4|N>=3}=79%	2*0.79
P{N>=5}=35.28%	P{N>=5|N>=4}=81%	2*0.81

P{N>=2}=73%,means at 73% moment, e(t) and e(t-1) have same sign; P{N>=3|N>2}=75%, can be explained as in condition that e(t) and e(t-1) have same sign, the probability of e(t+1) keeping the same sign is 75%. As shown in table 2, if at several continuous moments e(t) have the same sign, the probability of e(t) keeping this sign at next moment is likely quite large. This phenomenon is partially due to the effect of smoothing filter, which makes fundamental traffic to have a variation trend relatively persistent. If during several continuous moment, e(t) has the same sign, factor sign(t) should take a relatively large value (>1); as default, e(t) and e(t-1) have different signs, sign(t)=1.

The conditional probability of e(t) keeping its sign can be estimated by method described in section 3.1, and this probability is used to calculate factor sign(t). In the following experiment, the value of sign(t) is obtained by the conditional probability multiplied by 2.

$$sign(t) = \begin{cases} 1 & n(t) = 1 \\ 2 * P\{N \geq n(t)+1 \mid N \geq n(t)\} & n(t) > 1 \end{cases} \quad (7)$$

Absolute value – factor value(t). Factor value(t) is decided by the absolute value of prediction error e(t). If taken value(t) = |e(t)|/ σe for adjustment quantity, the LMS prediction delay will be effectively reduced at burst moment. However, too large |e(t)| will lead to violent fluctuation and larger error on prediction. So it is necessary to give an upper limit for factor value(t). According to Fig.2, P{|e(t)|<1.5*σe}>0.9 (σe can be estimated by dynamic method in section 3.1). If taken 1.5*σe(t) as the upper limit, we can probably avoid the prediction fluctuation of few too large |e(t)|, and exert the adjustment function of |e(t)| at most moments. A simple way to calculate value(t) is to choose the less one between 1.5*σe(t) and |e(t)|/ σe, as (8)

$$value(t) = \min(|e(t)| / \sigma_e(t),\ 1.5 * \sigma_e(t)) \quad (8)$$

Correction of LMS prediction. For one-step prediction, correction is made by adding the product of adjustment quantity ε(t) and standard deviation σ(t), to the LMS prediction result za1(t+1). The corrected value, noted as zb1(t+1), is the one-step EaLMS prediction result. Here, to multiply σ(t) corresponds to the normalization before applying LMS algorithm in 3.1.

$$zb1(t+1) = za1(t+1) + \sigma(t) \cdot \varepsilon(t) \tag{9}$$

As for multi-step prediction, such as two-step, at t moment, with the reception of measured traffic z(t), we get e(t), but not e(t+1). Weighted average of adjustment quantity ε at several previous moments can be used as an estimation of ε(t+1). For example, the adjustment quantity for two-step is,

$$zb2(t+2) = za2(t+2) + \sigma(t) \cdot \hat{\varepsilon}(t+1),\ \hat{\varepsilon}(t+1) = [2 \cdot \varepsilon(t) + \varepsilon(t-1) + \varepsilon(t-2)]/4 \tag{10}$$

4 Analysis of Experiment Results

Applying LMS and EaLMS predictor to LBL-PKT-4 and LBL-TCP-3 fundamental traffic, the comparison includes: 1. global performance: the comparison of global prediction error; 2. local performance: comparison of prediction delay and prediction error, (local performance is the magnification of global prediction fragment); 3. learning speed of adaptive predictor. For both methods, μ=0.01, smoothing filter degree is 10, and prediction error is calculated by root mean square error (RMSE).

4.1 Global Prediction Performance

Prediction interval of LBL-PKT-4 is (500,3600), and LBL-TCP-3 (500,7000). The reason that prediction interval starts from 500s is to ensure the learning process period of LMS. Due to the long duration of prediction interval, prediction curves cannot be distinguished from figures, but the prediction error shown in Table 3 and Table 4 indicates that EaLMS efficiently reduces global prediction error.

Table 3. RMSE of LBL-PKT-4 global prediction (10^3)

Step size	1	2	3
LMS	4.5780	7.3849	10.187
EaLMS	3.3341	5.9890	8.5731

Table 4. RMSE of LBL-TCP-3 global prediction (10^3)

Step size	1	2	3
LMS	4.4011	7.3301	10.337
EaLMS	3.2809	6.0019	8.7105

4.2 Local Prediction Performance

Local comparison is to examine the response speed, i.e. prediction delay of predictor, especially at the traffic burst moments. In fig.3, from above to bottom, prediction step sizes in (a), (b), (c) are respectively 1(s), 2(s), 3(s); z, zax, zbx respectively denote real traffic, LMS prediction value and EaLMS prediction value.

LBL-PKT-4 prediction interval (1180, 1280). Measured traffic with two successive bursts and a flat period after the first burst. According to fig.3 and table 5, EaLMS method reduces prediction delay and prediction error at the same time, but at the moments where traffic changes its variation trends (around 1203s, 1224s, etc.), EaLMS predictor has a larger error, especially in multi-step prediction, which is due to the inertial effect of adjustment.

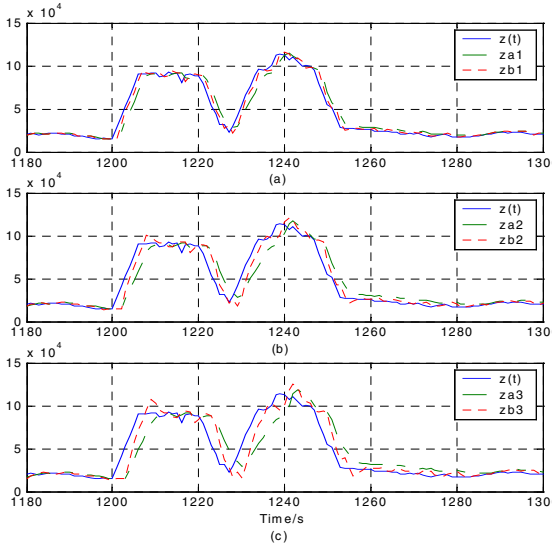


Fig. 3. Local prediction performance of LBL-PKT-4

Table 5. RMSE of LBL-PKT-4 local prediction (10^3)

Step size	1	2
LMS	8.5963	13.511
EaLMS	5.0464	9.1726

4.3 Learning Curve

Learning curve is defined as the variation relationship between mean square error (MSE) and iterative time n, which indicates the learning speed of adaptive forecasting method. Here calculating single learning curve of one-step prediction MSE makes the

comparison. Fig.4 illustrates that EaLMS prediction not only reduces prediction error, but also makes some improvement in learning speed.

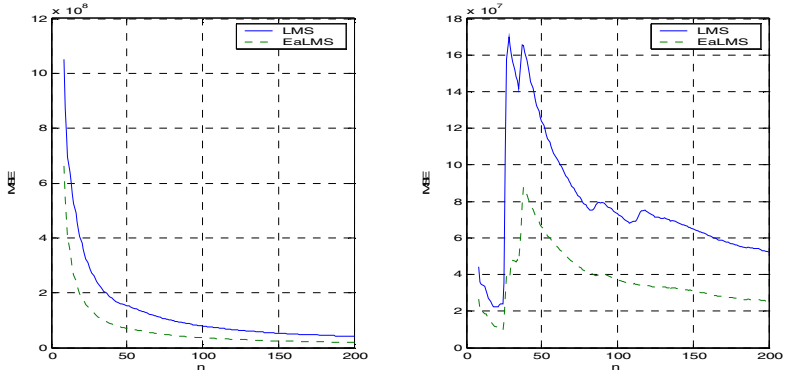


Fig. 4. One-step prediction learning curves of LBL-PKT-4 (left) and LBL-TCP-3 (right)

5 Influence of LPF on Traffic Prediction

The above experiments are made on fundamental traffic, i.e. the traffic obtained by passing smoothing filter. The influence of LPF (low-pass filtering) on traffic prediction is discussed in [11]. In general, LPF slows down traffic changes by removing short-term variations from the traffic. Therefore the filtered traffic has a more predictable behavior. However, high-frequency traffic also included important information about network behavior. It is necessary to select appropriate LPF as well as to search effective method to forecast high-frequency traffic.

The influence of smoothing degree k on one-step prediction error is shown in table 6 and table 7. Prediction interval: LBL-PKT-4 (500, 3600), LBL-TCP-3 (500, 7000). The effects of LPF on LMS and EaLMS are similar: the larger the smoothing degree is, the more stable the fundamental traffic is, and the better the prediction result is.

Table 6. Influence of smoothing degree on prediction error LBL-PKT-4 (10^3)

Degree	k =5	k =10	k =20
LMS	7.1255	4.5780	2.9027
EaLMS	6.3922	3.3341	2.0125

Table 7. Influence of smoothing degree on prediction error LBL-TCP-3 (10^3)

Degree	k =5	k =10	k =20
LMS	7.0784	4.3560	2.7289
EaLMS	6.1068	3.2535	1.9250

6 Conclusion

The paper makes tentative research for real-time traffic prediction. On the basis of LMS method, the authors bring forward EaLMS predictor. The main idea of EaLMS is to estimate traffic variation trend, and to correct LMS prediction value by an adjustment quantity. The adjustment quantity needs to insure the reduction of prediction delay at traffic burst moment, and should not cause strong fluctuation. Experiments of real traffic data shows that, compared with standard LMS method, EaLMS has efficiently reduced prediction delay at burst moment, decreased prediction error, and also improved the learning speed of prediction algorithm.

Compared with VSS-LMS, EaLMS does not change step size, so there is not a problem of influence on convergence. Calculation of the adjustment quantity uses statistics of traffic, and theoretically, it is suitable for other signal. In practice, combination of VSS-LMS and EaLMS may get better prediction performance. The computing process of EaLMS predictor does not need preexisted traffic data, and can be executed simultaneously with traffic measuring. This makes it significant for real-time prediction of network traffic.

References

1. N. K. Groschitz, G. C. Polyzos: A time series model of long-term NSFNET backbone traffic, In Proceedings of the IEEE International Conference on Communications (ICC'94), (1994) 1400-1404
2. E. S. Yu, C. Y. R. Chen: Traffic prediction using neural networks, In Proc. IEEE Globecom '93, (1993) 991-995
3. A. A. Tarraf, I. W. Habib, T. N. Saadawi: ATM multimedia traffic prediction using neural networks, In Proceedings of Global Data Networking, (1993) 77-84
4. Y. Liang, E. W. Page: Multiresolution Learning Paradigm and Signal Prediction, IEEE Transactions on Signal Processing, (1997) 2858-2864
5. A. Adas: Using Adaptive Linear Prediction to Support Real-Time VBR Video Under RCBR Network Service Model, IEEE/ACM Transaction on Networking, (1998) 635-644.
6. S. Chong, S. Li, and J. Ghosh: Predictive Dynamic Bandwidth Allocation for Efficient Transport of Real-Time VBR Video over ATM, IEEE JSAC, (1995) 12-23.
7. A.Adas: Supporting Real Time VBR Video Using Dynamic Reservation Based on Linear Prediction, IEEE Trans. Signal Processing, (1996) 1156-1167
8. X. Wang, Jung, Souhwan, J.S Meditch: Dynamic bandwidth allocation for VBR video traffic using adaptive wavelet prediction, in Proc. IEEE ICC'98, (1998) 549–553.
9. R. Wong, E. Johston: A variable step size LMS algorithm. IEEE Trans. on Signal Processing. (1992)
10. The Internet Traffic Archive: http://ita.ee.lbl.gov/
11. A. Sang, S. Li: A predictability analysis of network traffic. In Proc. IEEE INFOCOM 2000, (2000) 342-351

Dynamic Configuration between Proxy Caches within an Intranet

Víctor J. Sosa Sosa, Juan G. González Serna, Xochitl Landa Miguez, Francisco Verduzco Medina, and Manuel A. Valdés Marrero

Centro Nacional de Investigación y Desarrollo Tecnológico
Interior Internado Palmira s/n Col. Palmira
Cuernavaca, Morelos, México
{vjsosa, gabriel, userlamix, fverduzco, valdescompany}@cenidet.edu.mx

Abstract. A proxy cache is used for storing frequently-required data by a client, decreasing the server response time. Another way of using proxy caches is forming a proxy cache cooperative group. There are times when proxy caches, which are located in the same Internet service provider network, could cooperate between each other, but they do not know of the existence of other caches. In this paper we are proposing a proxy caches auto-configuration protocol that allows a proxy cache to detect other active caches. When it enters the intranet it detects them, and parameters are sent in order to do an auto-configuration. Another function that the protocol does is the cache monitoring of messages flowing between each other, with the purpose of temporarily or permanently deactivate deficient caches. That is why this protocol helps us to avoid possible-cooperative proxy caches under-utilization and avoids time loss when deficient proxy caches exist.

1 Introduction

Due to the increased number of users, as well as information, that the World Wide Web has had in the last decade, many problems have appeared, such as servers overload, congested networks, the increased use of available band width, to mention a few. Therefore, in order to decrease those problems, several propositions have been made. One of those propositions is the use of proxy caches in the Web, which consists on temporarily store near the client his frequently-required data, with the purpose of reducing the time of access to the information, decreasing the network congestion, and decreasing the server overload.

The proxy caches routes the communication toward Internet, like the firewalls. The proxy cache can intercept HTTP request done by clients. For instance, if a client does an information request, the proxy cache intercepts it, searching first that information in its memory. If it does not find it, the proxy cache forwards the request to the server of origin in representation of the client. The server answers to the proxy cache, which holds a copy in its memory and then sends this information back to the client [1, 2].

In order to use a proxy cache, we need to configure both the client (browser) and the proxy cache. In case there is more than one proxy cache, we need to configure

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 137–146, 2004.
© Springer-Verlag Berlin Heidelberg 2004

each proxy cache to indicate to them which ones it is going to work with. The purpose is to form a group of cooperative proxy caches. Also we have to indicate to the client which of the proxy caches it is going to work with.

The groups of proxy caches are used to make proxy caches work together when affinity groups exist, in such a way that they share documents stored in their caches, thus decreasing time response. The purpose of these proxy caches is to collaborate together to improve the service provided for the set of connected clients, preventing the requests forwarding to the server of origin.

This paper describes a system of cooperative proxy caches that are configured in a dynamic fashion, with the purpose of cooperating together. This system is implemented like an extension of freeware software Squid [3, 4], running under the Linux platform.

Squid is a proxy cache server that permits the use of only one connection towards Internet for all hosts. In addition, the proxy cache server Squid stores clients' frequently-required Web objects in its hard disk, in order to reduce the amount of communication packages flowing throughout the network, decreasing the network congestion, and decreasing the server overload.

2 Present Proxy Cache Scenario

The use of the proxy caches has been increasing due to their time response decrease when users with liking affinity exist and make the same requests, because proxy caches have the function to store frequently-asked information required by clients. Likewise, the use of the proxy caches has been increasing because the proxy caches can make cooperative groups, which share the information that is stored within theirs caches.

Nowadays, in order to make a group of proxy caches communicate between each other, each proxy cache has to do a manual configuration, and in this configuration have to explicitly specify which of the proxy caches it is going to work with. If, at some moment, a proxy cache does not work, the actives proxy caches continue sending request to this proxy cache, and a larger delay in answering the request occurs and it is necessary that a person manually discharges this proxy cache in each one of the active proxy caches. In the event that a proxy cache wants to be a member of the group, it is necessary to manually register this proxy cache in every active proxy cache so that it can work with the group.

On the other hand, there are proxy caches inside a same intranet or inside a same Internet Service Provided, and they could cooperate, but they do not know about the existence of others proxy caches, therefore we have a misuse of those proxy caches.

3 Proposed Scenario Involving Cooperative Proxy Cache Groups

In order to realize an automatic configuration between the active proxy caches that are inside a same intranet and the proxy cache that wants to be a member of that group, a protocol that does this function has been designed. Another function that the protocol

does is to discharge the proxy caches when they are not working properly or when their use is not viable.

The proxy cache that wants to be a member of a group will only have to connect itself to the network, and the proxy cache will be configured automatically in order to work with the existing active proxy caches. The active proxy caches detect this proxy cache and they will work together with it.

We have used the freeware Squid, which is a proxy cache server and can work with several protocols for establishing communication between the proxy caches, as well as forming hierarchies of proxy caches.

This protocol configures the proxy cache Squid that wants to be a member of the group, forming a cooperative proxy cache group. This protocol sends the necessary parameters so that this new proxy cache enters the group.

On the other hand this protocol has also the function of doing cache monitoring to detect when a proxy cache is not working or its efficiency is low, so it can discharge that proxy cache in all active proxy caches.

4 Auto-configuration Protocol between Proxy Caches

This protocol has the purpose of making discoveries of new proxy caches that want to be members of a group. To do that, the protocol establishes the necessary communication between the proxy cache that wants to be a member and the active proxy caches that are inside an intranet. Next, it configures this proxy cache that wants to be a member and reconfigures all active proxy caches in order to work with the new proxy cache. On the other hand, if any of the proxy caches is not working properly, it will be automatically discharged in each one of the active proxy caches. When a proxy cache requests success rate for another proxy cache is low, temporarily that proxy cache will be automatically discharged in the proxy cache that makes the requests, all of this with the purpose of stop asking this proxy cache for some time.

The protocol work begins when a new proxy cache is connected to a given network. The proxy cache that wants to be a member of that network sends several broadcast messages. In these messages it requires to become a member, and then requests the necessary parameters for auto-configure itself, as shown in Figure 1.

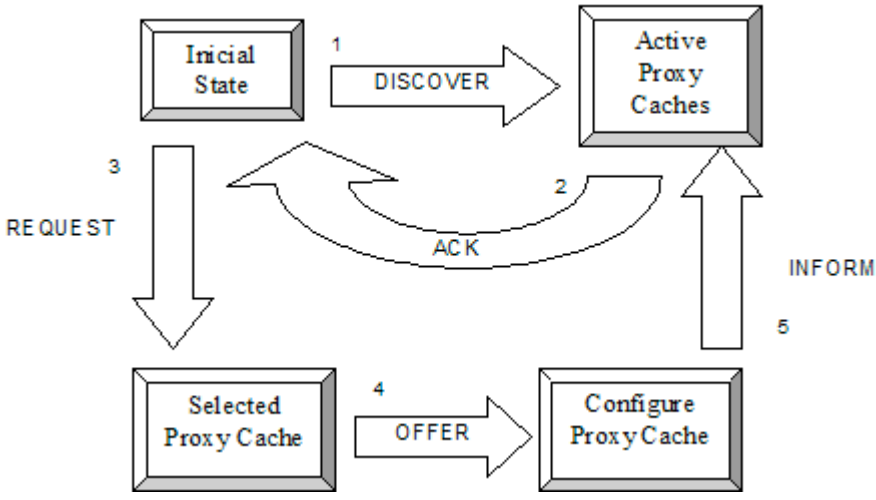


Fig. 1. The proxy cache that wants to be a group member sends a (*DISCOVER)* message to detect the active proxy caches. This proxy cache expects to receive a (*ACK*) message from each active proxy cache. Then it requests the necessary configuration parameters sending the (*REQUEST*) message to the first proxy cache that responds to this proxy cache. The selected active proxy cache sends the parameters by the (*OFFER*) message. Finally, the proxy cache that wants to be a group member, sends the (*INFORM*) message to the active proxy caches so that they can add this proxy cache to the group

During the dialog process between the proxy caches, the messages that are shown in Table 1 are produced.

Table 1. It shows the messages produced to realize the discovery of active proxy caches inside an intranet, to request the auto-configure parameters for a proxy cache that wants to be a group member and to inform active proxy caches of the adding of a proxy cache to the group

Messages	Description
DISCOVER	Broadcast call sent from the proxy cache that wants to be a group member for every active proxy cache of the group. It contains the IP address and membership type that the proxy cache will have.
ACK	Sent from the active proxy caches towards the proxy cache that wants to be member of a group with their IP addresses.
REQUEST	Sent from the proxy cache that wants to be a group member towards one active proxy cache, requesting the configuration parameters.
OFFER	Sent from one active proxy cache as response to the REQUEST message, offering the configuration parameters.
INFORM	Broadcast call sent from the proxy cache that wants to be a group member, indicating to all active proxy cache that they can add it to the group.

When a proxy cache that wants to be a member, sends a *DISCOVER* message as broadcast, that message has the membership type that the new proxy cache will have, as well as its IP address.

The server or servers that are active in that moment, when the *DISCOVER* message is received, will send an *ACK* message to the proxy cache that wants to be a member, indicating to it that the servers received the message, sending back theirs IP addresses. The active proxy caches will wait a certain amount of time for the proxy cache indication that it is configured, and the active proxy caches can add it as a member of their group.

The proxy cache that wants to be a member, when receiving one or more *ACK* response messages, will select the first proxy cache that responded with the *ACK* message. Then the aspiring proxy cache will establish a connection with the selected proxy cache and will send a *REQUEST* message to this proxy cache, requesting the necessary configuration parameters.

When the selected proxy cache receives a *REQUEST* message, it will send an *OFFER* message to the proxy cache that wants to be a member with the configuration parameters. Those parameters are: the port where the proxy cache Squid is working, the ICP (Internet Cache Protocol) [5, 6] protocol port, the active proxy caches IP addresses and their membership type. When the selected proxy cache finishes sending the parameters, the connection will end.

When the new proxy cache is configured, it will send an *INFORM* message as broadcast in order to indicate the active proxy caches that they can add this proxy cache to the group. This configured proxy cache will be in the same state as the other proxy caches (waiting for any proxy cache that wants to be a member).

In this moment proxy caches are configured in order to cooperate between each other. Now the active proxy caches are just monitoring the requests that the other proxy caches do and the responses that they send back.

The proposed protocol is working with the ICP protocol, which is supported by Squid. Our protocol monitors the protocol ICP HIT and MISS responses to requests for each proxy cache working with our protocol. If for every 100 continuous requests, the 90% of those are responded with MISS by a given proxy cache, the proxy cache that does the requests temporarily discharges that proxy cache for a certain amount of time. This time is given in seconds, being the default 3600 seconds. When that time passes by, the proxy cache reactivates the discharged proxy cache, and now the number of request decrease to 90. Now if for every 90 continuous requests, the 90% are responded with MISS, it is discharged again, but now the time is increased twice, 7200 seconds, and so on until it is definitively discharged of the proxy cache doing requests, because its use is no longer viable.

On the other hand, the ICP protocol helps our protocol to do the monitoring of proxy caches not working properly (it is not consider the case when a proxy cache has been discharged just for another proxy cache), so that they get automatically discharged in all active proxy caches.

With this we obtain a more complete protocol, which automatically registers and discharges proxy caches without the existence of a person modifying the configuration of each proxy cache when some change happens.

5 Results

This section shows some comparative analyses of two cooperating proxy cache systems, one cooperative caching system with manual configuration and other cooperative caching system with dynamic configuration. The cooperative proxy caching scenarios are built with six proxy caches connected by an intranet Ethernet 100mb. Each cache has: 2 Ghz Intel processor, 512MB RAM memory and a hard disk with 60MB. We have simulated a scenario with six proxy cache of the Spanish academic network backbone managed by the Center of Super Computing of Catalonia (CeSCa). We use workloads that were obtained from a cache located at CesCa. The simulation stops when it finishes replaying all the workload. Table 2 describes the workload. We decided to use this workload because it represents a real environment.

Table 2. Shows the workload characteristics. The workloads were obtained from a cache located at the Spanish academic network backbone managed by CeSCa

Number of requests	3,089,592
Number of Web objects	212,352
Number of clients (approx.)	11,765
Average requests per second	21
Transferred bytes	30 GB
Duration	2 days

Figure 2 shows the global results after replaying all the workload. As we can see in Figure 2a the global hit ratio of both configurations is almost the same. The slight difference is because the dynamic configuration sometimes could discharge one cache that is not longer viable. Meanwhile, that cache could receive a useful web object and lose the opportunity to get a HIT from it. Messaging passing is an important concern for caching communication that helps to find web objects into the collaborative caching group and at the same time allows caches activity monitoring. Figure 2b shows the total sum of bytes consumed in the intracache network. As we can see, dynamic configuration produces less bandwidth consumption. This situation is because when a proxy cache has been discharged, it no longer receives requests, reducing the number of messages. Figure 2c shows the total sum of response time after replaying all the workload. Dynamic configuration shows a better performance versus a manual configuration. When we are using manual configuration and a web object is requested, a package is sent to all the caches. This produces more load in each cache. This load impacts the response time of each cache. However, the dynamic configuration, many times can avoid sending messages to all caches, because a cache can know which one of the caches has the requested web object. This information could be obtained from its directory.

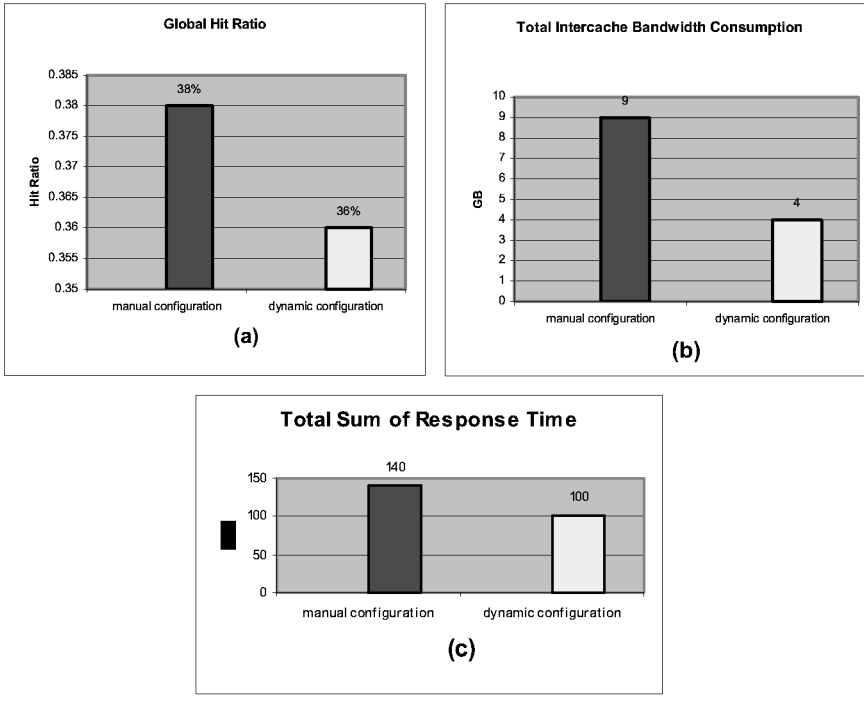


Fig. 2. Shows the global results after replaying all the workload

6 Related Works

In [7] the construction of adaptive caches is proposed, which consists on a group of proxy caches working in a cooperative manner, sharing objects between each other. This work uses the multicast, proposing several proxy caches groups, and so they can send the same document to several proxy caches that are inside the same group. This is also used when a host wants to request for an object, doing the request in a multicast form towards a specific group. If in this group the information is not found, the request is done to the server of origin, which is in another group. To communicate with the server of origin, one proxy cache of the group where the request is done is automatically auto-configured, in order to cooperate with the group where the server of origin is. However, this kind of configuration has the difficulty of the existence of administrative limits, and a world is foreseen where proxy caches are placed either in the network access point or inside several autonomous networks inside Internet.

In [8] a distributed multicast Web cache is described, which has a module for the auto-configuration of affinity groups. The hierarchies of proxy caches are auto-configured as the affinity Web pages are cached. This project has a LPC, a proxy cache which is in several places of the network. LPC has two components, a proxy cache server (called pump) and a distributed set of proxy cache clients (called filters).

The pump proxy cache monitors the access to the server, builds and destroys multicast channels with affinity groups. The filters act like only one virtual proxy cache. These filters do the monitoring of those channels with the purpose of finding affinity groups. Those affinity groups can be determined by a number of factors, which are: a-posteriori analysis of correlated retrieval, syntactic analysis of embedded HREF, analysis of semantic groups. However, in this work, the proxy caches are reconfigured are already installed in this network, modifying only the hierarchy depending on how the affinity groups get created, but neither aggregates nor takes away any proxy cache in an automatic manner.

In [5] and [6] the protocol *ICP (Internet Cache Protocol, RFC2186, RFC2187* for application*) version 2.0* is described, which does the messages interchange between proxy caches using UDP, in order to ask between each other about the existence of some URL (or Web object). Usually it is used to avoid having to go to the server of origin. With this protocol, we can get the accessibility of the proxy cache neighbors, handle the load balance for very busy proxy caches, and modify the wait time between messages. If after some continuous requests any of the proxy cache neighbors do not answer, it is marked as fallen. This protocol can deliver *ICP* requests to a multicast address and it can join the proxy cache neighbors to the multicast group in order to receive those requests. But in order to use *ICP* it is necessary that the proxy caches are correctly configured for cooperating together, so that this protocol does not have any auto-configuration module.

In [9] the protocol *HTCP* (*Hyper Text Caching Protocol, RFC2756*) *version 0.0* is described, which handles the discovery of HTTP proxy caches and data stored in the proxy cache memory by means of message interchange whether it is done by UDP datagrams or by TCP. It monitors the activity of the proxy caches. The difference between the protocol *ICPv2* and the protocol *HTCP* is that the protocol *HTCP* includes the complete header in the request as well as in the response. This protocol does the monitoring of information additions and deletes stored in the caches. It handles several header types, to know that one or more proxy caches store a copy of the Web object; to handle the caching policies of Web objects that include the details about the Web object; and for locally modifying the caching policy of the Web object. As in *ICP* protocol, it is necessary that the proxy caches are already configured so they can cooperate.

In [10] the protocol *WCCP* (*Web Cache Coordination Protocol*) *version 1.0*, which is a property of Cisco, is described. This protocol is used for associating only one router with one or more proxy caches (called proxy cache farm), with the purpose of discovering, verifying and informing the connectivity and availability of one or more proxy caches. The router transparently redirects the HTTP traffic towards a Cisco cache motor, with the purpose of not having to configure the clients. This protocol has a designated cache that indicates the router how to distribute the traffic for transparently redirect it through the associate proxy caches. The designated proxy cache uses a Hash table for mapping the IP destiny address of a package and redirects it towards the IP address of a proxy cache that is inside the farm. A proxy cache can join a farm by unicast messages sent to the router, but it is necessary to know the IP address for joining the group and so that proxy cache will begin to work inside the farm.

In [11] the protocol *WCCP* (*Web Cache Coordination Protocol*) *version 2.0* is described. This version allows the interaction between one or more routers and one or more proxy caches. The purpose is to establish and keep the transparent redirection of selected types of traffic as well as to distribute the load throughout a group of routers.

The designated proxy cache indicates the router or routers how the redirected traffic must be distributed between the members of the farm. A remission method is added which is used for sending redirected packages from a router towards a proxy cache. Another method that is added is a package return method which is used for returning redirected package from a proxy cache towards a router for a normal remission. It gives support for sending multicast messages between proxy caches and routers. This protocol can redirect traffic that is not HTTP. It allows proxy caches to refuse a redirected package and send it back to a router to be remitted. The method that is used to return the package towards a router is negotiable.

The *CARP* protocol (*Cache Array Routing Protocol*) *version 1.0* that is described in [12] is used to divide URLs spaces among a poorly coupled proxy server array by a Hash function. Another function that this protocol uses is a route function. This protocol includes a table of participant members or memberships (cooperative proxy cache group). It permits the tracking of the current state of an array table, and it also permits the dynamic load balance in the multiple servers that are inside of an array. This protocol stores the data of how much time a proxy server has been in its current state and how long it has been a member of this table. It handles a total load factor for an array that has to be handled for any member of this array. We can decide the cache size that is going to have any member of an array, but each member of the array has to be explicitly specified.

7 Conclusions

With the use of proxy caches it is not always necessary to go to the server of origin because documents previously requested are stored in the proxy cache. With this the network traffic gets reduced, the work load for the server of origin gets decreased, and the amount of communication packages that flow in the network gets decreased.

In case we have several persons that want to access the same pages, they will get an almost immediate response with the required documents, because these documents are stored within the cache of the proxy cache and it is no longer necessary to go to the server of origin.

On the other hand, when several proxy caches are joined together, they will have better performance, because proxy caches share the documents that they have stored in their caches, and so they decrease the response time. The ultimate goal of these proxy caches is to collaborate among each other in order to improve the service speed offered to the set of clients that visit the same pages when they are connected to the same network, but currently it is necessary to make a manual configuration of each proxy cache, indicating each one with which other proxy cache it is going to cooperate.

With this work, it is no longer necessary to make a manual configuration of each proxy cache that is inside the same autonomous system (intranet, Internet Services Provided network) in order to cooperate between each other. With the proposed protocol, when a proxy cache that wants to become a member is connected to the network, the protocol will detect the active proxy caches and automatically will add the new proxy cache in order to form a cooperative proxy cache group. Also, this protocol does the cache monitoring, so that it detects when some proxy cache is not working properly, discharging it in all active proxy caches. Likewise, if its request success

rate for another proxy cache is low, the proxy cache that does the request will temporarily discharge the faulty proxy cache. Therefore, it is not necessary to have a person that makes a manual configuration for each proxy cache when a proxy cache gets registered or discharged in the group, improving the services that proxy caches offer.

Now we are working on the auto-configuration of the client (browser). When a client enters the network, it will automatically get configured, and it will get assigned to an active proxy cache. If in a given moment this proxy cache fails, the client automatically gets assigned to another proxy cache. All of this happens transparently for the client, i.e., the client does not know what is happening or what changes are being made.

With the new part of the project that we are working on, we are doing a more complete system so that it is not necessary to configure either the client or the proxy cache, achieving that everything gets done automatically, improving the service speed.

References

1. Sosa, V. J.: Arquitectura para la Distribución de Documentos en un Sistema Distribuido a Gran Escala. Ph.D. Thesis, Centro de Investigación en Computación, IPN. (2002)
2. Sosa, V. J., Navarro, L.: Arquitectura para el acceso a documentación distribuida basada en Caches en el Web. Universidad politécnica de Cataluña (2000)
3. Moisés, A.: Squid bajo Windows NT. Online documentation, available at http://www2.idesoft.com/squid/que.htm, IDESOFT (2003)
4. Wessels, D.: Squid Web Proxy Cache. Online documentation available at http://www.squid-cache.org (2003)
5. Wessels, D., Claffy, K.: Application of Internet Cache Protocol (ICP), version 2. UCSD / National Laboratory for Applied, Network Research (1997)
6. Wessels, D., Claffy, K.: Internet Cache Protocol (ICP) version 2. UCSD / National Laboratory for Applied, Network Research (1997)
7. Zhang, L., Floyd, S., Jacobson, V.: Adaptive Web Caching. UCLA / Computer Science Department (1997)
8. Touch, J., Hughes, A. S.: The LSAM Proxy Cache - a Multicast Distributed Virtual Cache. USC / Information Sciences Institute (1998)
9. Vixie, P., Wessels, D.: Hyper Text Caching Protocol (HTCP/0.0) (2000)
10. Cieslak, M., Forster, D.: Web Cache Coordination Protocol V1.0. Cisco Systems (1999)
11. Cieslak, M., Forster, D., Tiwana, G., Wilson, R.: Web Cache Coordination Protocol V2.0. Cisco Systems (2000)
12. Valloppillil, V., Ross, K. W.: Cache Array Routing Protocol v1.0. Microsoft Corporation / University of Pennsylvania (1998)

A Market-Based Scheduler for JXTA-Based Peer-to-Peer Computing System

Tan Tien Ping[1], Gian Chand Sodhy[1], Chan Huah Yong[1], Fazilah Haron[1] and Rajkumar Buyya[2]

[1]School of Computer Science
Universiti Sains Malaysia
11800 Penang, Malaysia
{tienping, sodhy, hychan, fazilah} @cs.usm.my
[2]Grid Computing and Distributed Systems Laboratory
Department of Computer Science & Software Engineering
University of Melbourne, Australia
raj@cs.mu.oz.au

Abstract. Peer-to-Peer (P2P) computing is said to be the next wave of computing after client-server and web-based computing. It provides an opportunity to harness a lot of idle peer-resources such as desktop computers across the Internet, for solving large-scale computing applications. Each peer is autonomous and it needs incentive for sustained contribution of its resources to P2P applications. In addition, a flexible and efficient job scheduling is needed to harvest the idle computing power as cheaply and economically as possible.
This paper introduces an economic based job scheduler for mapping jobs to resources in P2P computing environment. The scheduler has been implemented with the Compute Power Market (CPM) system developed using Sun JXTA P2P technology. Our scheduler can be configured depending on users' quality of service requirements such as the deadline and budget constraints. Our scheduler follows a hierarchy scheme. The design allows multiple consumers and multiple providers to schedule and run jobs. To allow wider support for a wide variety of applications, the system is designed to allow easy 'plug in' of user applications.

1 Introduction

P2P computing has been touted as the next wave of computing after client-server and web-based computing [1][22]. The P2P computing paradigm harnesses resources such as storage, computing cycles, contents and human presence that are available at the edge of the Internet. An advantage of P2P computing model is that everyone can contribute their resources while being autonomous. Another advantage of P2P is that a lot of otherwise unused resources can be harvested for the development of science, engineering, and business. Different types of P2P systems have been developed to support file sharing, distributed computing, collaboration, searching, instant messaging and mobile devices. Example systems and applications [22] include Napster, ICQ, Jabber, Gnutella, FreeNet and SETI@Home [14].

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 147–157, 2004.
© Springer-Verlag Berlin Heidelberg 2004

P2P computing systems aim to exploit the synergies that result from co-operation of autonomous peers. For this cooperation to be sustainable, peers (resource contributors) need an incentive [8][12][15]. Efforts such as SETI@Home [6] are successful in attracting a large number of contributors due to (a) their exciting application theme—search for an extraterrestrial intelligence— and (b) their objective of sharing results with the public. There is a huge potential in creating and transforming P2P networks into a computing marketplace that brings together providers and consumers. In such marketplace, peers (providers) gain economic incentive by providing access to their resources; and they also get encouraged to offer value-added services. The consumers benefit by gaining access to large-scale resources on demand and having an ability to select resources by based on their quality of service requirements.

In [18], we proposed a market-based resource management and job scheduling system, called Compute Power Market (CPM), for P2P computing on Internet-wide computational resources as market-based systems offer economic incentive for resource providers and also support the regulation of supply-and-demand for resources [8][11][15]. The CPM primarily comprises of markets, resource consumers, resource providers and their interactions. Over the last three years [19], our team has carried out an implementation of CPM using Sun's JXTA [16] P2P computing framework. It supports various economic models for resource trading and matching service consumers and providers. The CPM components that represent markets, consumers and providers are Market Server, Market Resource Agent, and Market Resource Broker (MRB).

This paper focuses on a market-based scheduler implemented as a component of CPM's MRB. The broker is responsible for providing master-worker style application scheduling services by discovering and selecting suitable resources that match user requirements. The CPM scheduler adopts scheduling algorithms, originally developed for Grid environments [13], and incorporates them within P2P computing-based CPM system.

2 Related Work

Although job scheduling on parallel and distributed systems has been investigated extensively in the past, they are limited to cooperative and dedicated environments or system-centric in nature. Moreover, scheduling in different environments such as P2P often involves different challenges and policies, for instance variation of resource availability with time and the presence of large-scale heterogeneity. A number of projects have investigated scheduling of computations on Internet-based distributed systems. They include AMWAT [4], Condor [20], XtremWeb [23], Entropia [24], AppLes[5], Nimrod-G[11], Javelin++ [10], and Java Market [8].

3 CPM and JXTA P2P Framework

JXTA is a open source project initiated by Sun Microsystems. One of its main objectives is to develop a standard protocol that all P2P applications can utilize, since there are a lot of P2P applications which cannot operate, interact and utilize each other's service. Worldwide developers have contributed to new services, for example file sharing chatting and others.

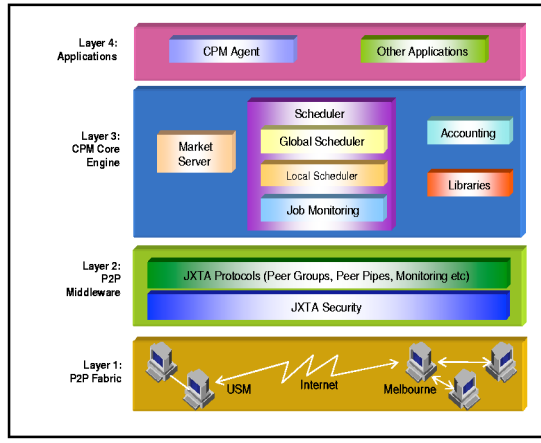


Fig. 1. CPM/P2P Framework.

Compute Power Market (CPM) is a market-based resource management service developed under JXTA. Through CPM, consumer and resource owner can trade computational resources over a P2P network. Figure 1 shows the components of CPM organized into layers. Layer 1 connects geographically distributed compute devices through JXTA network across the Internet. Layer 2 is the middleware comprising JXTA protocols and security. The Core Engine layer contains the main CPM modules, while the application layer contains programs that utilize the functionality of CPM modules.

The scheduler interacts with other modules in CPM to achieve the overall objective. A global scheduler of CPM Market Resource Broker (MRB) interacts with the CPM Market Agent to retrieve available resources in the market. With this information, the global scheduler can then schedule tasks to selected resources. Local scheduler will then schedule the execution of the tasks. Global scheduler and local scheduler will provide an accounting module and billing information, which will be used to bill the customer accordingly.

4 Scheduler Architecture

The scheduler is one of the main components in CPM. It is responsible for assigning tasks from customer to resource providers. The scheduler will be operating in a market environment, where it schedules tasks based on deadline and budget. The type of jobs which can take advantage of our scheduler are those that can be partitioned into smaller tasks and executed in an independent manner such as Monte Carlo simulation, image processing applications, molecular docking and others.

In general, schedulers can be categorized according to their scope. They are the global scheduler and the local scheduler. The global scheduler is also known as macro-scheduler, while local scheduler is also called a micro-scheduler [3]. The global scheduler decides the resource to which the user job is to be allocated based on a certain global scheduling policy, but local scheduler chooses the job to be run on the local system based on the local system policy.

Schedulers can be categorized according to their architectural model. The three commonly used models for organisation and structuring of schedulers are centralized, hierarchical, and distributed models [2][21]. The CPM scheduler follows a hierarchical model. The main components of a CPM scheduler are global scheduler, job monitoring, resource discovery and local scheduler.

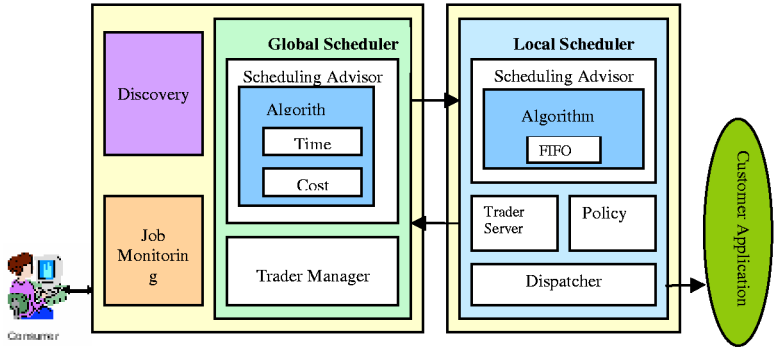


Fig. 2. CPM scheduler architecture.

4.1 Scheduler Components

Global scheduler is responsible for the scheduling of a customer's job. The components of a global scheduler are:

- Scheduling Advisor – Schedule tasks to local scheduler based on a specific scheduling algorithm.
- Scheduling Algorithm – Two types of heuristics scheduling algorithms are supported. They are Cost Optimized and Time Optimized scheduling algorithms [13]. Cost Optimized algorithm schedules tasks for customer where their main concern is cost, while making sure that deadline is reached. Time Optimized

scheduling tries to complete the tasks as soon as possible and within the specified cost.

- Trader Manager – Negotiate price with Trader Server at local scheduler for an agreeable price.

Job Monitoring module monitors the status of tasks and triggers the global scheduler to reschedule if failure occurs. This includes tasks which do not execute or complete within expected period of time or tasks which have failed. The discovery component is used for searching available resource providers over the network.

The CPM local scheduler can accept tasks from more than one global scheduler but only one task is executed at a time. The components of a local scheduler are:

- Scheduling Advisor – It accepts or rejects a given task based on resource provider's specified scheduling policies. It schedules the execution of tasks based on local scheduling algorithm.
- Scheduling algorithm – Support scheduling based on first-in-first-out (FIFO) method.
- Trader Server – Negotiate price with Trader Manager.
- Policy – User can specify the scheduling policy which decides what tasks get accepted and what task does not. Three types of policies are supported: minimum price of the total given tasks, total maximum time length acceptable by resource provider and availability.
- Dispatcher – Dispatches customer's application for execution. Two types of application are supported currently. They are java applications and Windows executable programs.

4.2 Scheduling Activities

Scheduling can be divided into 5 steps. The steps involved are:

i) Resource discovery, where resources are searched for and found.
ii) Resource trading, which involves retrieving the necessary information for selecting and scheduling tasks at the next stage.
iii) Scheduling, where tasks are matched to resources, using a scheduling algorithm.
iv) Execute and monitor the task on resources used.
v) Rescheduling: It handles reassignment of failed tasks or schedule variation to due to the change in the availability of resources.

4.2.1 Resource Discovery

Before a customer can schedule tasks to peers, resources need to be discovered. In JXTA, resources, services and others are represented as advertisements. The CPM resource advertisements contain information such as price, speed of computation and others, which are published by resource providers. There are 3 ways to discover advertisements, either through local cache, direct discovery or indirectly through a rendezvous peer [16]. A Rendezvous peer is a special peer, which provides the service of discovering other peers or resources.

4.2.2 Resource Trading

The trading module in the global scheduler and the local scheduler is used for negotiating agreeable price. In the current implementation, there is not much interaction involved. We use a flat-price market model whereby providers advertise their resources using resource advertisements. Global scheduler that discovers the resource advertisements will then decide which peer to hire depending on a few criteria, such as cost, speed and completion time. The information of cost and speed can be retrieved from the resource advertisement. However for the completion time, the global scheduler needs to query the local schedulers. Communication between global scheduler and local scheduler is done via JXTA pipe service. The benefit of using pipes for communication is that it hides the need to know each other's IP address and port number, using only peer IDs. Besides that, the communication can cross firewalls. These are important aspects, especially in P2P environment.

An "order ID" is also being returned when a global scheduler queries for the completion time at a local scheduler. The order ID is used to resolve contention when two or more global schedulers are interested in a peer at the same time. When a global scheduler wants to make a purchase, it needs to supply the order ID, which will change when an order is successfully made.

4.2.3 Scheduling

The global scheduler allocates tasks to a local scheduler through a contract. A contract is an agreement between the global scheduler and the local scheduler on the award of one or more tasks to it. Each task can be different in terms of functionality, runtime, input files, etc. However, they must not have any dependencies among them. A contract received by a local scheduler is put into a queue, and processed in FIFO order. A task from the contract will be processed. The required files, such as application and data files, will be fetched when the task is about to run.

A resource is selected based on the resource's price versus customer's budget and resource's completion time (next available time) versus customer's deadline. We provide two scheduling algorithms for a user to choose, depending on their priority in cost and speed [12]. In cost optimization scheduling, the scheduler will try to schedule as many tasks as possible to the cheapest provider as long as deadlines are not exceeded. As for time optimization, the scheduler will schedule to make sure tasks are completed as soon as possible, taking care that budget is not exceeded.

4.2.4 Execution and Monitoring

When a task is ready for execution, the local scheduler will request the necessary data or files from the global scheduler. After the required data is available, the task can be executed. The module that is responsible for the execution of customer applications is the Dispatcher. The dispatcher learns of the type of application and then executes the right command to run the application. The current implementation supports 2 types of applications, Java programs and Windows executable programs.

Job monitoring module is responsible for monitoring the status of the tasks. When a task fails to execute or complete within expected period, job monitoring module will abort the task and reschedule it to another peer. To avoid the possibility of a failed

task being moved from one peer to another, user can set the maximum number of times a task can be rescheduled.

4.2.5 Rescheduling

Scheduler will reschedule a task when it is triggered by the job monitoring module. Rescheduling a task is just like scheduling, where it is reassigned to a new provider. When the scheduler cannot find a suitable resource, customers may update their QoS parameters such as budget and/or deadline constraints, so that resources that meet the new criteria can be found.

5 Scheduling Experiments and Results

We have developed a simple image processing application as a simulation. The application is used for segmenting an object from an image using adaptive thresholding method [17].

The objective of the experiments are to see how the scheduler schedules a task under different budget and deadline constraints. We conducted our test in a local area network (LAN) with bandwidth of 100Mbps. Table 1 shows the hardware used We have configured a job consisting of 40 tasks on a Window machine to be scheduled to 4 provider machines (labeled as Win2000, XP1, XP2 and Linux), the specifications of which are shown in Table 1. Each task uses an input image involving an estimated 28141 million operations, which will be processed by an image processing application as shown in Figure 3. All machines are benchmarked using a standard application where the number of floating point operations they can perform in a second is known. The price is set as tokens per hour and the value of tokens can be mapped to a real currency. Each task consists of an image with the same dimension and parameters. We then schedule the tasks out to resources where objects in the image are segmented and returned to the customer.

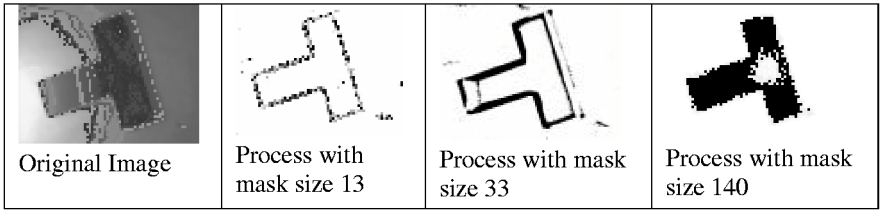


Fig. 3. Processed images with different mask size which have been resized.

In the first experiment, we have allocated 500 tokens with the deadline of 1 hour to perform the job, which consists of 40 tasks. We select Time Optimization as the scheduling algorithm to schedule the jobs. Figure 4 shows the task completion time by the machines. From the graph we can see that all tasks completed before the deadline (i.e. 3600 seconds). Machines XP1 and XP2 have been scheduled the most

Table 1. CPM testbed and hardware configuration

PC	Specification			Description			Label
	OS	Memory (GHz)	RAM (MB)	Role	Benchmark (mflops)	Price (token/hour)	
1	Win2K	1	256	Provider	130	80	Win 2000
2	WinXP	2.4	512	Provider	180	300	XP2
3	WinXP	2.4	512	Provider	180	250	XP1
4	RedHat	0.486	256	Provider	50	40	Linux

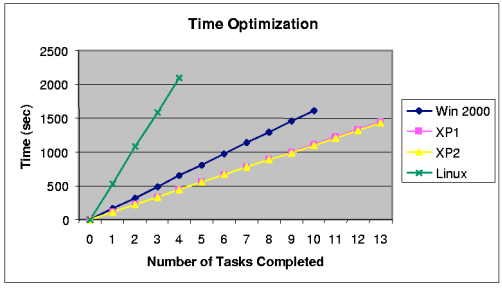


Fig. 4. Graph showing Time Optimization scheduling of 40 tasks. Budget allocated is 500 tokens with time deadline of 1 hour.

tasks (13 tasks), since both have the highest processing speed and the budget allocated is sufficient. Therefore, in cases where budget is more than required, cost becomes a less important constraint, and the dominant factor which determines the completion time is the number of resource providers and their respective processing speeds.

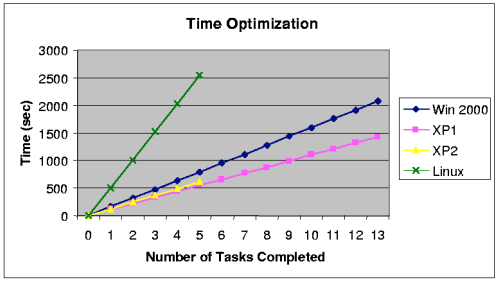


Fig. 5. Graph showing Time Optimization scheduling of 40 tasks. Budget allocated is 350 tokens with time deadline of 1 hour.

In the second experiment, we still use Time Optimization algorithm to schedule the same jobs, but now the budget has been reduced from the previous 500 tokens to 350 tokens. From Figure 5, we can see that reducing the budget will reduce the number of tasks allocated to more expensive machines. In this case the number of tasks allocated to machine XP2 has been reduced to 5. Cheaper but slower machines like Win2000, for instance, get more tasks.

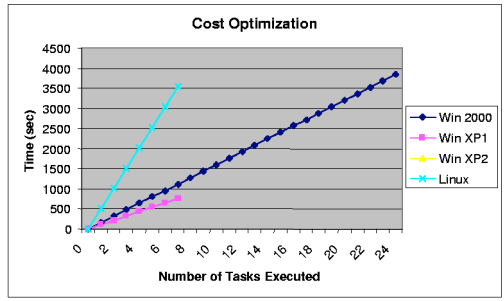


Fig. 6. Graph showing Cost Optimization scheduling of 40 tasks. Budget allocated is 300 tokens with time deadline of 1.5 hour.

For the final experiment, we use the same job again for our test. However, we change our scheduling strategy to Cost Optimization, allocating only 300 tokens, with the duration of 1.5 hours. From Figure 6, we can see that the scheduler utilizes the cheapest resources by allocating as many tasks as possible to them, for instance machines Win2000 and Linux. The most expensive resource, machine XP2, does not get any task at all. Notice that all 40 tasks are completed well before the deadline (of 5400 seconds).

From this experiment, we can deduce that if task execution time can be reasonably estimated, then the Cost Optimization scheduling algorithm provides better results in terms of budget used and completion time (within the deadline given).

6 Conclusion

We discussed a market-based scheduler for JXTA-based P2P computing system. It followed hierarchical model for system architecture. It consists of global scheduler, which is implemented as part of the CPM market-resource broker and local scheduler, which implemented as part of the CPM market resource agent. We discussed in depth the implementation of two market-based global scheduling algorithms within CPM global scheduler along with experimental and performance results.

Acknowledgements. A financial assistance from the Ministry of Science, Technology and Environment of Malaysia (MOSTE) is gratefully acknowledged. We thank Rob Gray (Monash University) and Srikumar Venugopal (University of Melbourne) for their valuable comments on the paper.

References

[1] Clay Shikey, "What is P2P… And What Isn't", http://www.openp2p.com/pub/a/p2p/2000/11/24/shirky1-whatisp2p.html, O'Reilly Network. Nov 2000.

[2] Vijay Subramani, Rajkumar Kettimuthu Srividya, Srinivasan P. Sadayappan, "Distributed Job Scheduling on Computational Grids using Multiple Simultaneous Requests", *Proceedings of the 11th IEEE International Symposium on High Performance Distributed Computing HPDC-11 20002 (HPDC'02),* Scotland, 2002.

[3] Steve J. Chapin and Eugene H. Spafford, "Support for Implementing Scheduling Algorithms Using MESSIAHS", *Scientific Programming*, 1994.

[4] Garry Shao, "Adaptive Scheduling of Master/Worker Applications on Distributed Computational Resources", *Ph.D. Thesis*, University of California, San Diego. May 2001.

[5] Francine Berman et. al. "Adaptive Computing on the Grid Using AppLeS", *IEEE Transactions on Parallel and Distributed Systems*, Vol. 14, No. 4 IEEE Press, USA, Apr 2003.

[6] David Anderson, Jeff Cobb, Eric Korpela, Matt Lebofsky, Dan Werthimer, "SETI@home: An Experiment in Public-Resource Computing", *Communications of the ACM,* Vol. 45 No. 11, ACM Press, USA, November 2002.

[7] Henri Casanova, Arnaud Legrand, "Heuristics for Scheduling Parameter Sweep Applications in Grid Environment", *Proceedings of the 9th Heterogeneous Computing Workshop (HCW'2000*). Cancun, Mexico, May 2000.

[8] Yair Amir, Baruch Awerbuch, and Ryan S. Borgstrom, "The Java Market: Transforming the Internet into a Metacomputer". *Technical Report CNDS-98-1*, Johns Hopkins University, 1998.

[9] Peter Cappello, Bernd Christiansen, Mihai F. Ionescu, Michael O. Neary, Klaus E. Schauser, and Daniel Wu, "Javelin: Internet-Based Parallel Computing Using Java", *Proceedings of the 1997 ACM Workshop on Java for Science and Engineering Computation*, June 1997

[10] Michael O. Neary, Sean P. Brydon, Paul Kmiec, Sami Rollins, Peter Capello, "Javelin++: Scalability Issues in Global Computing", *Future Generation Computing Systems Journal*, Vol.15(5-6):659-674, Elsevier, Netherlands, 1999.

[11] Rajkumar Buyya, David Abramson, Jonathan Giddy, "Nimrod/G: An Architecture for a Resource Management and Scheduling System in a Global Computational Grid", *Proceedings of 4th International Conference on High Performance Computing in Asia-Pacific Region (HPC Asia 2000)*, Beijing, China, 2000.

[12] Rajkumar Buyya, "Economic-based Distributed Resource Management and Scheduling for Grid Computing", *Ph.D. Thesis*, Monash University Australia, April 2002.

[13] Rajkumar Buyya, Jonathan Giddy, and David Abramson, "An Evaluation of Economy-based Resource Trading and Scheduling on Computational Power Grids for Parameter Sweep Applications", *Proceedings of the 2nd Workshop on Active Middleware Services (AMS 2000),* Kluwer Academic Press, Pittsburgh, USA, August 1, 2000.

[14] W. T. Sullivan, D. Werthimer, S. Bowyer, J. Cobb, D. Gedye, D. Anderson, "A new major SETI project based on Project Serendip data and 100,000 personal computers", *Proceedings of the 5th International Conference on Bioastronomy*, 1997.

[15] Carl A. Waldspurger, Tad Hogg, Bernado A. Huberman, Jeffrey O. Kephart and Scott Stornetta, "Spawn: A Distributed Computational Economy", *IEEE Transactions on Software Engineering*, IEEE Press, USA, February 1992.

[16] Brendon J. Wilson, *JXTA*, New Riders Publishing, Indiana, June 2002.

[17] Anatol Piotrowski and Sivarama P. Dandamudi, "Performance Sensitivity of Variable Granularity *Proceedings of International Conference on Massively Parallel Computer Systems*, Colorado Springs, April 1998

[18] Rajkumar Buyya and Sudharshan Vazhkudai, "Compute Power Market: Towards a Market-Oriented Grid", *Proceedings of the First IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGrid 2001),* Brisbane, Australia, May 15-18, 2001.

[19] Rajkumar Buyya Fazilah Haron Chan Huah Yong, "CPM on Jxta", http://compute-power-market.jxta.org/

[20] Matt Mutka and Miron Livny, "Scheduling Remote Processing Capacity In A Workstation-Processing Bank Computing System"*, Proceedings of the 7th International Conference of Distributed Computing Systems,* September 1987.

[21] Rajkumar Buyya, David Abramson, and Jonathan Giddy, "An Economy Driven Resource Management Architecture for Global Computational Power Grids", *Proceedings of the 2000 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2000)*, Las Vegas, USA, June 26-29, 2000.

[22] Andy Oram (ed), *Peer-to-Peer Harnessing the Power of Disruptive Technologies*, O'Reilly Press, USA, 2001.

[23] Cecile Germain, Vincent Neri, Gille Fedak and Franck Cappello, "XtremWeb: building an experimental platform for Global Computing", *Proceedings of the 1st IEEE/ACM International Workshop on Grid Computing (Grid 2000)*, Bangalore, India, Dec. 2000.

[24] Andrew Chien, Brad Calder, Stephen Elbert, and Karan Bhatia, "Entropia: architecture and performance of an enterprise desktop grid system", *Journal of Parallel and Distributed Computing*, Volume 63, Issue 5, Academic Press, USA, May 2003.

Reducing on the Number of Testing Items in the Branches of Decision Trees[1]

Hyontai Sug

Division of Internet Engineering, Dongseo University
Busan, 617-716, South Korea
sht@Dongseo.ac.kr

Abstract. Even though decision trees are one of the mostly used data mining methods, the cost of testing long branches is a hindrance for usability of the method, if some feature values in the branches require high costs to get or are not available. As a method to overcome this difficulty, we applied a multidimensional association rule algorithm with some restriction to the branches of generated decision tree, and found that most of the branches in the decision tree have shorter and more reliable multidimensional association rules as subsets of the branches so that reducing the number of testing items may be possible. Therefore, by using the found shorter and reliable rules, costs related to testing each item in the branches can be reduced.

1 Introduction

The target domain of data mining and knowledge discovery in databases (KDD) contains a tremendous amount of data so that the task of data mining is comparable to that of finding a gemstone in the sand. Decision trees, one of the most important data mining technique, have been very successful in prediction tasks, so finding decision trees with possibly the smallest error rates for a given data set has been a major task for researchers. A single tree is very good for a hierarchical view of target data thus leads to easy understanding of the discovered knowledge. So, that's the reason why decision trees are favored by many people. But, it is well known that building an optimal decision tree is NP complete problem, so we use a greedy algorithm to genenate a decision tree in reasonable time. But greedy algorithms have their own weak points; when we build a decision tree, the root node of each subtree is chosen among the attributes which have not yet been chosen by ancestor nodes so that the selected attribute is the best split based on some criteria for the split.
Even though KDD problems usually contain a tremendous amount of data, decision trees cannot fully take advantage of the abundance of training data. As a tree is being built, each branch starts having less objects, so the reliability of each branch becomes

[1] This work was supported by Dongseo University, “Dongseo Frontier Project” Research Fund of 2003.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 158–166, 2004.
© Springer-Verlag Berlin Heidelberg 2004

worse than the upper branches. Therefore, a single tree may lead to unnecessary tests of features and may not represent rules that are best for some substantial subset or collection of the objects in the database. So, in this paper, we want to propose a method that not only utilizes conventional decision trees for them to be more useful, but also can provide an improved cost model for the application of the target database, when we have enough training data.
In section 2, we first briefly provide related work to our research, and in sections 3 and 4 we present our method in detail and show the result of experiment. Finally section 5 provides some conclusions.

2 Related Work

Decision tree algorithms are based on greedy method. So, generated trees are not optimum and some improvement may be possible. There has been much efforts to build better decision trees. For example, one of standard decision tree algorithm C4.5 [1] uses entropy-based measure, and CART [2] uses purity-based measure, and CHAID [3] uses chi-square test-based measure for split. And also there has been scalability-related efforts for large databases such as SLIQ [4], RainForest [5], and SPRINT [6]. SPRINT tries to solve the scalability problem by building trees in parallel.
There has been also a lot of research for feature subset selection and dimensionality reduction problem. One of major problem in scientific data is that the data are often high dimensional so that it takes very long computing time for pattern recognition algorithms. To solve this problem dimensionality reduction algorithms have been invented to select the most important features so that further processing like pattern recognition algorithms can be simplified without comprising the quality of final results. Commonly used approach is PCA(Principal Component Analysis) and MDS(Multi-Dimensional Scaling) which find subset of features based on mutually-orthogonal linear combinations of the original features [7]. On the other hand, feature subset selection algorithms try to eliminate irrelevant features from attribute set [8, 9, 10]. Irrelevant features are features that are dependent on other features so that they have bad effects on their size and error rate of generated trees. If there are n features, we can have 2^n possible feature subsets, so feature subset selection is a hard problem. There are two directions to deal with the computation problem--wrapper approach and filter approach. The wrapper approach adds features to the set of good features incrementally based on test results of underlying algorithms. It needs $O(n^2)$ runs of the algorithm to test. The filter approach uses some heuristic to select good subset of features, so it's faster than the wrapper approach. Most dimensionality reduction algorithms are kinds of filter approach.
Even though we have selected appropriate features, decision tree algorithms have two congenital problems; fragmentation and unnecessary test problem by disdaining minority data. Fragmentation problem occurs, because each node in the tree has fewer training examples. Unnecessary tests at the lower levels of the tree occur, because minor feature values are hardly selected as upper nodes in the tree. For example, we

can decide HIV patient is sick, but HIV is minor compared to other disease, so it may be tested at the lower levels of the tree. As a result, we may have to do many unnecessary tests.

Another related work is association rule finding algorithms. There are many good algorithms to find association rules efficiently. For example, a standard association algorithm, Apriori, large main memory-based algorithm like AprioriTid [11], hash-table based algorithm, DHP [12], random sample based algorithms [13], tree structure-based algorithm [14] or even a parallel version of the algorithm [15]. There are also multidimensional association rules. Multidimensional association rules are basically an application of general association rule algorithms to table-like databases [16]. In papers like [17, 18], multidimensional association rules have better accuracy than decision trees for most of example data. The data used for the experiment are from UCI machine learning repository [19]. They also used the found association rules and other prediction methods in combination for better prediction. But large-size data were not used for comparison due to time complexity.

Even if decision trees have some congenital problems, they are one of the mostly used data mining methods, because they have many good characteristics, especially the easy-to-understand structure. So, when there are a lot of training data, we want to appreciate the structure of decision trees, as well as to provide more economical way of prediction by providing shorter rules of reliability of each branch of decision trees, which is the contribution of this paper. Testing less number of features is very important especially for medical domain, because some tests require very high costs or some test items are not available at all. The difference of our approach from feature subset selection or dimensionality reduction algorithms is that our method is applied for each branches of decision tree, and can be applied feature-selected decision trees also. Moreover, our method appreciate the structure of the original decision trees.

3 The Method

Let **H** be the set of possible hypotheses and m be the number of training examples in the training set. The sample size m in PAC (Probably Approximately Correct)-learning theory [20] for a rule of reliability can be represented by $m \geq (1/\varepsilon) \bullet (ln(1/\delta) + ln\ |\mathbf{H}|)$, where ε and δ are small constants. In other words, a hypothesis that is consistent with at least m training examples has error at most ε with probability at least $1 - \delta$. So, based on $|\mathbf{H}|$, we may set m as a minimum support number with some small values for ε and δ for reliable rules. Because we are dealing with large databases, which means we have a lot of training examples, it is easy to set m for reliable rules. The following is the procedure of the method:

1. Determine a minimum support number m based on the PAC.
2. Generate a decision tree and determine interval structures for continuous features.
3. Discretize continuous features based on the found interval structures and convert the decision tree into rules.

4. Run a multidimensional association rule finding algorithm with the minimum support number m.
   -Depending on available computing resources, one may find frequent itemsets of length of one less than the longest branch in the generated decision tree.
5. Generate rules.
6. For each branch in the tree, find the subset of the branch in the found multidimensional association rules such that the rules are subsets of the branch with better confidence.

Because we want to find better rules for each branch in the generated decision tree, discretization in step 3 is based on the interval structure in the generated decision tree. The time complexity to find short association rules in step 4 is almost linear to the size of databases, and it takes less time to generate rules from the found frequent itemsets than conventional association rule algorithms, because class feature is fixed.

4 Experiment

4.1 Decision Tree Generation

To see the effect of the method for very large data sets, 'census-income' in UCI machine learning repository was used. It has 299,285 examples and among them 199523 examples are for training and 99762 examples are for testing. There are 41 features and 8 of them are continuous-valued features. Class probabilities for class values -50000 and 50000+ are 93.8% and 6.2% respectively. We used C4.5 [1] to generate a decision tree. Because the data set has continuous features, it took very long time to generate a tree. So, 1/12 size sample is used to generate a tree. Decision trees in other sample size were too big for comprehension or not good enough in error rates. The generated tree has 214 nodes with expected error rate of 0.053. It took more than 8 hours to generate a tree for the sample with Sun Blade1000 workstation.
A branch in the generated tree by C4.5 looks like "(capital gains ≤ 6849) and (dividends from stocks ≤ 0) ➔ -50000 (14645.0/511.9)." The left and right number in a parenthesis represent the number of cases and upper limit of expected misclassification cases based on U_{CF} respectively. The pessimistic error rate of the rule, or a leaf, based on the U_{CF} (E, N) is (N-E)/N where E is the expected number of error cases, N is the number of cases that is in the leaf, and CF is 0.25 as a default. The confidence based on this pessimistic error rate has the property that it becomes larger than just using the number of error cases as N becomes smaller. Fig.1 shows the values of pessimistic error rate as N changes compared to the confidence in association rules when it is a fixed value of 0.95. So, confidence values for each individual leaf in decision tree of C4.5 are more exaggerated when the number of training examples are smaller.

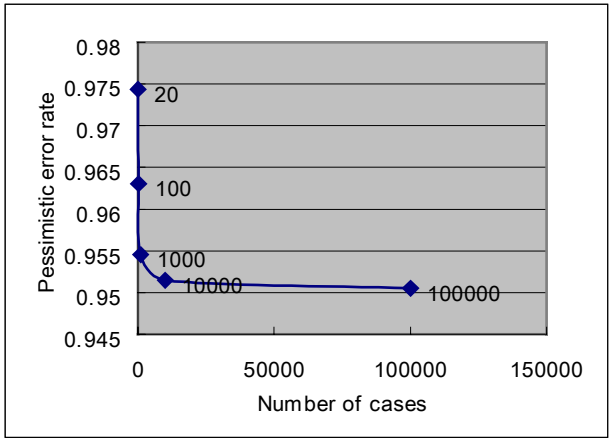


Fig. 1. The Distribution of Pessimistic Error Rates corresponding to 0.95

4.2 Discretizing Continuous Features and Converting to Rules

Among eight continuous features, only six features are appeared in the generated tree. So, the used six continuous features are converted to nominal values as follows. Each value is a partitioning point.

Table 1. The partitioning points for continuous features

Feature	Partitioning points
Capital gains	6849, 743, 14344
Capital loses	880, 1876
Dividends from stocks	0, 456, 500, 903, 1150
Weeks worked in year	44, 51
Number of persons worked for employer	0, 1
Age	29, 36, 48

The generated decision tree is converted to rules using the above intervals for the next step. Total of 2681 rules are generated. The distribution of rules in the decision tree is summarized in table 2. The average length of decision tree rules is 7.58.

When a tree is converted rules, a branch in the original tree may be converted to several rules. For example, suppose we have nominal value {a1, a2, a2+} for interval structure a1≤, a2≤, >a2. If a branch in the tree is (a2≤A) and (B=b1) ⇒ d1, which is translated to (A={a1, a2}) and (B=b1) ⇒ d1 in the interval structure, then we can have two rules, (A=a1) and (B=b1) ⇒ d1, (A=a2) and (B=b1) ⇒ d1. See figure 2 for reference.

Table 2. The distribution of rules in the decision tree

Length of rules	Total number of rules
2	1
3	69
4	52
5	33
6	20
7	626
8	1680
9	200

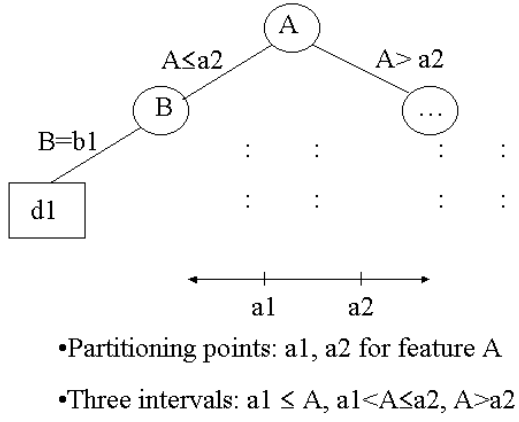


Fig. 2. An example decision tree

4.3 Finding Subsets of Rules in Each Branch

According to PAC learning theory after partitioning, $|\mathbf{H}|$ becomes 2.41216×10^{36} for the data set, so when we set ε and δ be 0.15, m can be 571 which is 0.2% of the data. Because we want to generate rules of enough support number and error rates of reliability, we use both of training and test 'census-income.data' file to find short association rules.

It took half an hour to find rules with rule length limit of 3. We found 12,975 rules of length 2 or 3 which are the subsets of decision tree rules and also have better confidence than the decision tree rules'. Average of 4.84 subset rules of better confidence are found by the multidimensional association rule finding algorithm for each decision rule in the tree. For example, the following is one of the 2681 rules. The rules imply that even though we have a branch with error rate of 0.947 in the decision tree, we have shorter rules in the branch by applying association rule discovery method.

Note also that each association rule has very large number of supporting cases so that the confidence of each rule is very reliable; ε and 1-δ value are 0.000491 and 0.999509 for 186,003 cases, and 0.00097 and 0.99903 for 93,447 cases. In other words, the first association rule has error at most 0.000491 with probability at least 0.999509, the third association rule has error at most 0.00097 with probability at least 0.99903,.

IF (capital_gains ≤ 6849) and (0 < dividend_from_stocks ≤ 456) and
(Capital_loses ≤ 880) and (weeks_worked_in_year ≤ 44) THEN
-50000 (.947 with support 691 cases)

The subsets found by association rule algorithm are the following three rules.

IF (capital_gains ≤ 6849) THEN -50000 (.948 with support 186003 cases, ε & δ = 0.000491)
IF (capital_gains ≤ 6849) and (capital_loses ≤ 880) THEN
-50000 (.953 with support 183356 cases, ε & δ = 0.000498)
IF (capital_gains ≤ 6849) and (weeks_worked_in_year ≤ 44) THEN
-50000 (.9511 with support 93447 cases, ε & δ = 0.00097)

The largest number of supporting cases for the subset rules is 183,356 and the smallest number of supporting cases for the subset rules is 586. The following table 3, and 4 summarize the result.

Table 3. Summary of found subsets of each rules in the decision tree

Total number of rules in the decision tree	Number of subset rules of better confidence	Average number of subset rules of better confidence
2,681	12,975	4.84

Table 4. Number of supporting cases of each rules in the decision tree

	Number of supporting cases	ε & δ
Maximum	183,356	0.000498
Medium	85,820	0.001056
Minimum	586	0.146

Note that each association rule has very large number of supporting cases so that the confidence of each rule is very reliable.

5 Conclusions

Testing less number of attributes is important especially for medical domain, because some tests require very high costs or some test items are not available at all. Even though decision trees are favored in many data mining fields including medical domain due to their easy-to-understand structure, decision trees have two problems; fragmentation problem of training data and the possibility of unnecessary tests for minority data. A single tree may lead to unnecessary tests of attributes and may not represent rules that are best for some substantial subset or collection of the objects in the database.

Association rule algorithms can find reliable rules exhaustively. Applying the algorithms to large databases may be prohibitive, if a given minimum support is small. But the algorithms can find reliable rules even for large databases in reasonable time, if we do not want to find association rules of all length.

We have devised a method that utilizes good points of decision trees (comprehensibility) and association rule algorithms (exhaustive rule finding) simultaneously. We applied multidimensional association rule finding algorithm with restriction in rule length to each branch of a decision tree, and found that most of the branches have better rules of shorter length. If the class of an instance can be predicted by shorter multidimensional association rules on the branch, testing cost can be reduced, since the evaluation of other feature values that are required by the corresponding path or branch in the decision tree are not required.

References

1. Quinlan, J.R.: C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, Inc. (1993)
2. Breiman, L., Friedman, J., Olshen, R. and Stone, C.: Classification and Regression Trees. Wadsworth International Group (1984)
3. StatSoft, Inc.: Electronic Statistics Textbook. Tulsa, OK: StatSoft. WEB: http://www.statsoft.com/textbook/stathome.html. (2004)
4. Mehta, M., Agrawal, R., and Rissanen, J.: SLIQ : A fast scalable classifier for data mining. (EDBT'96), Avignon, France, (1996)
5. Gehrke, J., Ramakrishnan, R., and Ganti, V.: Rainforest: A framework for fast decision tree construction of large datasets. Proc. 1998 Int. Conf. Very Large Data Bases, pages 416-427, New York, NY, August 1998.
6. Shafer, J, Agrawal, R., and Mehta., M.: SPRINT : A scalable parallel classifier for data mining. Proc. 1996 Int. Conf. Very Large Data Bases, 544-555, Bombay, India, Sept. 1996.
7. Jolliffe, I.T.: Principal Component Analysis. Springer Verlag, 2nd ed. (2002)
8. Almuallim, H., Dietterich, T.G.: Efficient Algorithms for Identifying Relavant Features, Proc. of the 9th Canadian Conference on Artificial Intelligence, 38-45 (1992)
9. Kononenko, I., et. al.: Overcoming the myopia of inductive learning algorithms with RELIEF, Applied Intelligence, Vol.7, no. 1, 39-55 (1997)

10. Liu, H., Motoda, H.: Feature Extraction, Construction and Selection: A Data Mining Perspective, Kluwer International (1998)
11. Agrawal, R., Mannila, H., Toivonen, H., Verkamo, A.I.: Fast Discovery of Association Rules, In Advances in Knowledge Discovery and Data Mining, Fayyad, U.M., Piatetsky-Shapiro, G., Smith, P., Uthurusamy, R. ed., AAAI Press/The MIT Press (1996) 307-328
12. Pak, J.S., Chen, M., Yu, P.S.: Using a Hash-Based Method with Transaction Trimming for Mining Association Rules, IEEE Transactions on Knowledge and Data Engineering, vol.9, no.5, (1997) 813-825
13. Toivonen, H.: Discovery of Frequent Patterns in Large Data Collections, phD thesis, Department of Computer Science, University of Helsinki, Finland (1996)
14. Han, J., Pei, J., Yin, Y.: Mining Frequent Patterns without Candidate Generation, SIGMOD'00, Dallas, TX, May 2000
15. Savasere, A., Omiecinski, E., Navathe, S.: An Efficient Algorithm for Mining Association Rules in Large Databases, College of Computing, Georgia Institute of Technology, Technical Report No.: GIT-CC-95-04 (1995)
16. Han, J., Kamber, M.: Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers (2000)
17. Li, W., Han, J., Pei, J.: CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules, Proceedings 2001 Int. Conf. on Data Mining (ICDM'01), San Jose, CA.
18. Liu, B., Hsu, W., Ma, Y., Integrating Classification and Association Rule Mining, Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98), New York, NY (1998)
19. Hettich, S., Bay, S.D.: The UCI KDD Archive [http://kdd.ics.uci.edu]. Irvine, CA: University of California, Department of Information and Computer Science (1999)
20. Russel, S., Norvig, P., Artificial Intelligence: a modern approach, Prentice Hall, Inc. (1995)

CORBA-Based, Multi-threaded Distributed Simulation of Hierarchical DEVS Models: Transforming Model Structure into a Non-hierarchical One

Ki-Hyung Kim[1] and Won-Seok Kang[2]

[1] Dept. of Computer Eng. Yeungnam University,
214-1 Daedong, Gyungsan, Gyungbuk, Korea
kkim@yu.ac.kr
http://nclab.yu.ac.kr
[2] Advanced Information Technology Research Center(AITrc), KAIST
373-1, Kusung-Dong, Yusong-Gu, Daejon, Korea
wskang@world.kaist.ac.kr

Abstract. The Discrete Event Systems Specification (DEVS) formalism specifies a discrete event system in a hierarchical, modular form. This paper presents DEVSCluster, a CORBA-Based, multi-threaded distributed simulation scheme for models specified by the DEVS formalism. The simulator transforms a hierarchical DEVS model into a non-hierarchical one. This transformation can ease the synchronization of the distributed simulation of DEVS models by enabling the transfer of events by direct remote method invocations, not explicit message transfers. By virtue of this feature, we can utilize CORBA for the event handling in DEVS-Cluster. To show the effectiveness of the proposed simulation scheme, we realize DEVSCluster in Visual C++, and conduct a benchmark simulation for a large-scale logistics system. We compare the performance of MPI and CORBA-based implementations. The performance result shows that the proposed methodology works correctly and performs better than the previous approaches.

1 Introduction

Discrete-event simulation is frequently used to analyze and predict the performance of systems. Simulation of large, complex systems remains a major stumbling block, however, due to the prohibitive computation costs. Distributed discrete-event simulation (or shortly, distributed simulation) offers one approach that can significantly reduce these computation costs.

Conventional distributed simulation algorithms usually assume the simulation consists of a collection of logical processes(LPs) that communicate by exchanging timestamped messages or events. The goal of the synchronization mechanism is to ensure that each LP processes events in time-stamp order; this requirement is referred to as the *local causality constraint*. The algorithms can

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 167–176, 2004.
© Springer-Verlag Berlin Heidelberg 2004

be classified as being either conservative[1] or optimistic[2]. Time Warp is the most well known optimistic method. When an LP receives an event with timestamp smaller than one or more events it has already processed, it rolls back and reprocesses those events in timestamp order.

Since distributed simulation deals with large and complex systems, the following issues should be addressed: model verification and validation, model reusability, and user-transparency of distributed simulation details, etc. The Discrete Event Systems Specification(DEVS) formalism, developed by Zeigler [3], is a formal framework for specifying discrete event models. The DEVS modelling and simulation approach provides an attractive alternative to conventional logical process-based modelling approaches used in distributed simulation by its set-theoretical basis, its independence of any computer language implementation, and its modular, hierarchical modelling methodology.

Distributed simulation of DEVS-based models differs from the conventional logical process-based distributed simulation [1] [2] in that: (i) the formalism differentiates external and internal events of the models; and (ii) for simulation of DEVS models, the hierarchical simulation mechanism has been used [3]. Owing to these differences, most of the distributed DEVS approaches have exploited only the specific parallelism inherent in the formalism. These can be broadly classified into two approaches: synchronous [4] which utilizes only the parallelism in simultaneous events and asynchronous [6] [3] [5] [7] which combines both the hierarchical simulation mechanism and distributed synchronization algorithms such as Time Warp [2].

This paper proposes a CORBA-based distributed simulation methodology for hierarchical, modular DEVS models, called DEVSCluster. The contribution of this paper is two folds; The first contribution is devising a non-hierarchical simulation scheme for DEVS simulation. It first transforms a hierarchical DEVS model into a non-hierarchical one, and then applies the simplified non-hierarchical simulation mechanism to the transformed model. By simplifying the model structure, DEVSCluster can be easily extended to the distributed version. It employs Time Warp as a basic synchronization protocol.

The second contribution of this paper is applying the distributed object technologies such as CORBA(Common Object Request Broker Architecture) [8] as an underlying communication mechanism which is particularly useful for the DEVS simulation since it is inherently an object oriented modelling formalism. DEVSCluster is the first implemented version of a CORBA-based distributed simulator for DEVS models. To fully utilize the capabilities of CORBA and prevent deadlock conditions, we have designed a multi-threaded version of DEVSCluster. It can also utilize the flexible extensibility based on the open industry standards, such as naming service, multi-threading, the support of heterogeneous platforms, dynamic invocation interface, and the utilization of the implementation repository for managing simulation libraries.

To show the architectural effectiveness of DEVSCluster, we design a model of a large-scale logistics system and perform experiments in a distributed environment. The results show DEVSCluster works correctly and outperforms the previous approaches. For the performance evaluation of the CORBA based dis-

tributed simulation, we compare performance results for both CORBA and MPI (Massage Passing Interface)[9].

The rest of this paper is organized as follows: Section 2 describes an overview of the DEVS formalism. Section 3 presents the proposed simulation methodology. Section 4 presents the performance results. Finally, Section 5 concludes the paper.

2 DEVS Formalism

The DEVS formalism is a sound formal modelling and simulation (M&S) framework based on generic dynamic systems concepts [3]. DEVS is a mathematical formalism with well-defined concepts of hierarchical and modular model construction, coupling of components, and an object oriented substrate supporting repository reuse. Within the formalism, one must specify (1) the basic models from which larger ones are built, and (2) the way in which these models are connected together in a hierarchical fashion. Top down design resulting in hierarchically constructed models is the basic methodology in constructing models compatible with the multifaceted modelling approach.

A basic model, called the atomic model (or atomic DEVS), specifies the dynamics of the system to be simulated. As with modular specifications in general, we must view the above atomic DEVS model as possessing input and output ports through which all interactions with the external world are mediated. To be more specific, when external input events arrive from another model and are received on its input ports, the model decides how to respond to them by its external transition function. In addition, when no external events arrive until the schedule time, which is specified by the time advance function, the model changes its state by the internal transition function and reveals itself as external output events on the output ports to be transmitted to other models. For the schedule time notice, an internal event(*) is devised as shown in the above definition.

Several atomic models may be coupled in the DEVS formalism to form a multi-component model, also called a coupled model. In addition, closed under coupling, a coupled model can be represented as an equivalent atomic model. Thus, a coupled model can itself be employed as a component in a larger coupled model, thereby giving rise to the construction of complex models in a hierarchical fashion.

Detailed descriptions for the definition of the atomic and coupled DEVS can be found in [3].

3 Non-hierarchical Distributed Simulation Scheme of DEVS Models

In this section, we propose a new simulation scheme for DEVS models, called DEVSCluster. Since the architecture of DEVSCluster has its own sequential simulation mechanism for DEVS models, we first present the sequential version

of DEVSCluster. After that, we describe the motivation and architecture of the distributed version of DEVSCluster.

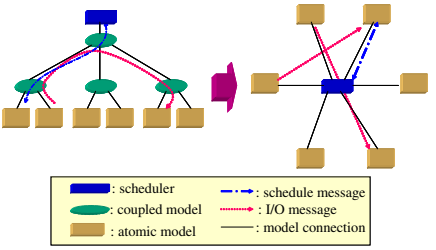


Fig. 1. Proposed simulation mechanism for DEVS models

3.1 Non-hierarchical Simulation of DEVS Models

The basic motivation of DEVSCluster is to simulate hierarchical DEVS models in a non-hierarchical manner. Fig. 1 shows the proposed mechanism DEVSCluster. We transform hierarchical DEVS models into a non-hierarchical form without any loss of information.

As described in the previous section, DEVS models have two types of models, coupled and atomic models. Coupled models specify the coupling between component models (which can be another coupled models, thereby forming recursively hierarchical model structure). The major functions of the coupled model and its associated abstract simulator, also called coordinator, are two-fold. The first function is event scheduling for its children models. The second function is the external event passing between component atomic models. We translate this information of coupled models into a flat-structured model information class, named ModelId, which is attached to the abstract simulator of atomic models. We now can remove the coupled models from the DEVS model. By this translation, hierarchical structure of the DEVS model can be flattened. Each atomic model has the information of the external input/output influence relationship which eventually come from the coupled models and can send directly external input/output events to the influencees by using the information.

Another important consideration factor is the internal event scheduling. In the conventional hierarchical scheduling, each coupled model and its associated simulator (called a coordinator) schedules the internal events only for its children component models. Thus, scheduling can utilize the locality inherent in the model hierarchy. Since there is no model hierarchy in DEVSCluster, a central scheduler should handle all events generated from all atomic models. To improve the scheduling efficiency of the central scheduler, we employ the SPLAY tree structure for the scheduler which can achieve an amortized log N performance for insertion and deletion of events. The use of the SPLAY tree can reduce the scheduling overheads of the central scheduler, due to the inherent locality of models.

3.2 CORBA-Based, Multi-threaded Distributed Architecture of DEVSCluster

Contrast to the previous simulation schemes that are hierarchical, DEVSCluster is basically non-hierarchical and thus can be implemented in a conventional simulator structure. In fact, D-DEVSim++[5], a hierarchical implementation of the optimistic distributed simulation mechanism for DEVS models, has to handle the transfer of all events between models through explicit messages, not method invocations. We will discuss this topic in detail in this section.

Fig. 2 shows the distributed version of DEVSCluster, and Fig. 3 shows the block diagram of DEVSCluster. Instead of using MPI-based explicit message queue for Time Warp synchronization, DEVSCluster can utilize CORBA as an underlying distributed object technology. This makes DEVSCluster an adequate simulation engine for heterogeneous network computing environment. Notice that in the figure, CORBA servant(*DEVSClusterManagerImpl*) invokes threads for incoming external messages to access models and simulators which are shared resources with the local simulation scheduler. One of the reason of this multi-threaded design of DEVSCluster is to prevent deadlock conditions which might occur if we don't use threads in processing external messages

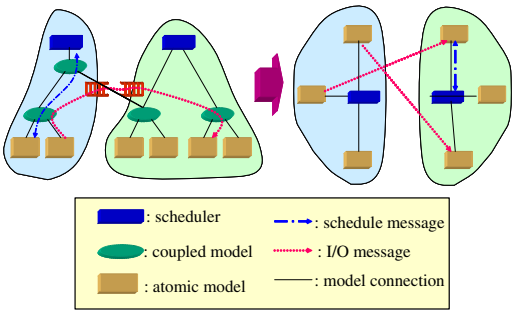


Fig. 2. Distributed simulation mechanism for DEVSCluster

Another important consideration is the method of event transfer. In the previous approach, each external input/output event transfer between models should be made of an explicit message regardless of whether the models reside in the same computer node or not. In contrast, DEVSCluster can utilize simple method calls between model objects. This can simplify the synchronization between event processing and increase the stability of the simulation engine. In fact, the larger the size of the model, the larger the amount of unsynchronized events in the event queue in the previous scheme. The queue should synchronize each incoming events with the existing ones. The synchronization job may involve complex rollback or annihilation, thereby becoming a bottleneck of the distributed processing. DEVSCluster eliminates this event queue, and a model sends/receives directly an input/output events to/from another model regard-

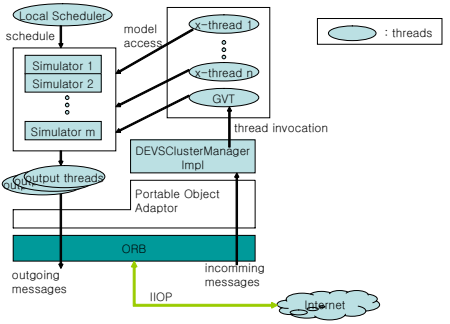


Fig. 3. Block diagram of DEVSCluster

less of whether they reside in the same node or not. The only required change of DEVSCluster compared to the sequential version is the algorithm of the individual abstract simulator of atomic models which implements distributed Time Warp synchronization.

The following shows the IDL codes for DEVSCluster. Through this standard interface description, DEVSCluster can be used in a heterogenous computing environment.

IDL Codes for DEVSCluster

```
module DEVSCluster {
        struct Commbuf{
            unsigned long       src;
        long long       time;
        unsigned long       dst;
        long long       tN;
        unsigned long       simcount;
        unsigned long       priority;
        unsigned long       dupcount;
        unsigned long       type;
        unsigned long       sign;
        unsigned long       mesgid;
        unsigned long       func;
        string          buf;
        };
        interface DEVSClusterManager{
        void RunSimulation();
        void SendToXMesgThread(in CommBuf buf);
        long long Cal_Lvt(in long long lvt);
        void SetGVT(in long long gvt);
        };
};
```

4 Experimental Results

To show the effectiveness of the proposed simulation scheme, DEVSCluster, we have conducted a benchmark simulation for a large-scale logistics system. In this section, we first describe the simulated logistics system and its simulation model. The simulation performance results are compared with the previous simulation engines.

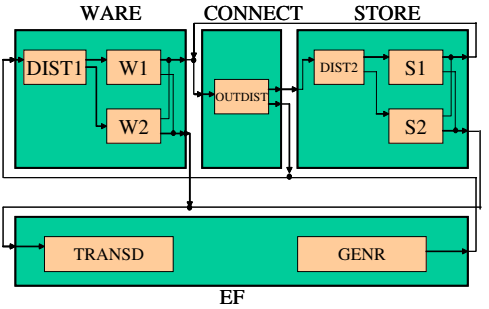


Fig. 4. Basic structure of the logistics system model

4.1 Logistics System

The automatic logistics system becomes more important as the size of the system becomes large and complex. For the help of the decision-making, simulation techniques can be used usefully. However, even in the normal sized logistics system, one of the important problems to be tackled is the excessive simulation execution time and the requirement of the large memory. In this case, distributed simulation can help the problem. Fig. 4 shows the basic structure of the logistics system model. In the logistics system, warehouses, stores, and control centers are located over wide areas: stores spend products, warehouses supply the products, and control centers control the logistics cars to chain the stores and warehouses. In this system, the objective is to minimize the number of the required cars while satisfying the orders of all stores. To find a shortest path between warehouses and stores is basically a travelling salesperson problem. We utilized the genetic algorithm for this problem. Also we abstracted the loading of products on cars as a bin packing problem and used the best-fit algorithm for this problem. Fig. 5 shows the Java-based GUI of the simulation system. The simulation system models major 4 cities of KOREA.

4.2 Results

We have conducted simulation experiments using DEVSCluster and the previous hierarchical simulation algorithm[5], called DDEVSim++. The configuration of

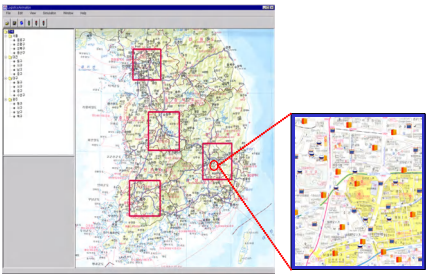


Fig. 5. GUI of logistics network simulation

the simulation platform is four Pentium IV-based Windows XP systems connected by the 100 Mbps Ethernet. We implemented the simulation system by Visual C++ 6.0 and the GUI by JAVA.

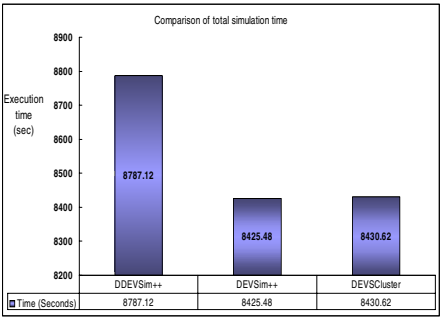


Fig. 6. Simulation execution time comparison of DEVSim++[3], DDEVSim++[5], and DEVSCluster

At first, we compared the sequential version of DEVSCluster to the sequential version of DDEVSim++. Fig. 6 shows the performance result. Basically DEVSCluster surpasses DDEVSim++ in performance. This result comes from the difference of the event handling method. DEVSCluster handles events by method calls of objects while DDEVSim++ handles events by explicit message passing.

For the test of the scalability of DEVSCluster, we experimented the simulation while changing the number of nodes. Fig. 7 shows the result. For 2 and 4 nodes, we compared three implementations of DEVSCluster: *MPI*, *CORBA-Non-Thread-Send*, and *CORBA-Thread-Send* which invokes threads for executing remote method calls. DEVSCluster achieves the nearly linear speedup on the left-hand-side figure. This result, of cause, comes first from the enough parallelism of the application compared to the distributed simulation overhead. The

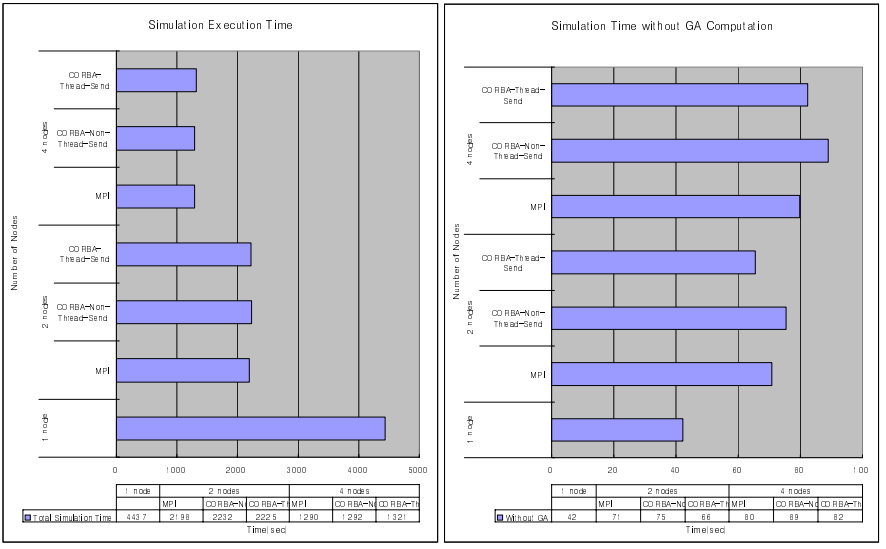


Fig. 7. Simulation time analysis of DEVSCluster

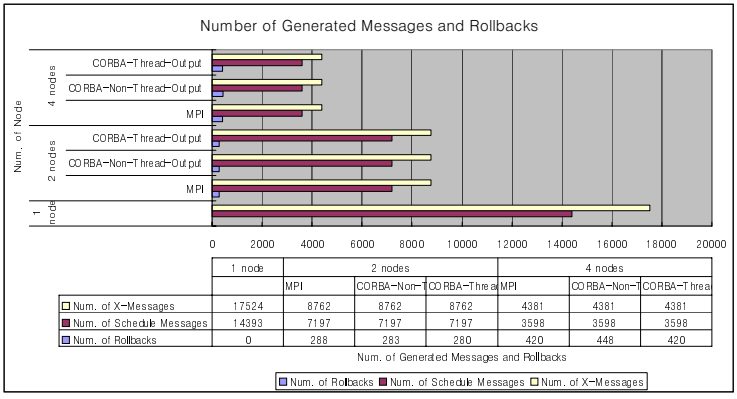


Fig. 8. Number of generated messages and rollbacks

right-hand side graph shows pure distributed simulation overheads of DEVSCluster. Notice that MPI is superior to the CORBA-based implementations in terms of communication speed. Another interesting result is that CORBA-Thread-Send outperforms CORBA-Non-Thread-Send. This is because CORBA-Thread-Send can utilize the buffering effect of threading.

Fig. 8 shows the number of generated messages and rollbacks while increasing the number of nodes. X-messages imply the external input/output messages. The result shows that the total number of committed messages for each distributed version is the same as the sequential version.

5 Conclusion

In this paper, we proposed a CORBA-based, Multi-threaded, distributed simulation methodology for the models specified by the DEVS formalism. The proposed methodology, named as DEVSCluster, basically transforms the hierarchical DEVS models into non-hierarchical ones. This can eliminate the overheads incurred during the conventional hierarchical simulation mechanism. It also simplifies the synchronization for distributed simulation. To show the efficiency of DEVSCluster, we performed a benchmark experiment for a large-scale logistics system. Due to its simple architecture, DEVSCluster outperformed the previous approaches. If we compare CORBA and MPI, MPI slightly outperforms CORBA. However CORBA-based implementation offers flexible extensibility based on the open industry standards.

References

1. Chandy, K.M. and Misra, J.: Distributed Simulation: A Case Study in Design and Verification of Distributed Programs, IEEE Trans. on Software Eng. vol. 5, no. 5, (1978) 440–452
2. Fujimoto, R.M.: Optimistic approaches to parallel discrete event simulation, Transactions of the Society for Computer Simulation International, vol. 7, no. 2, October, (1990)153–191
3. Zeigler, B.P., Praehofer, H., and Kim, T.G.: Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems, 2nd Ed., Academic Press, (2000) 261–287
4. Chow, A.C.: 1996. Parallel DEVS: A parallel, hierarchical, modular modeling framework and its distributed simulator. Transactions of the Society for Computer Simulation International vol. 13, no. 2, pp.55-67
5. Kim, K.H., Seong, Y.R., Kim, T.G., and Park, K.H.: Distributed Simulation of Hierarchical DEVS Models: Hierarchical Scheduling Locally and Time Warp Globally. Trans. of SCS vol. 13. no. 3, (1996) 135–154
6. Kim, K.H., Seong, Y.R., Kim, T.G., and Park, K.H.: Ordering of Simultaneous Events in Distributed DEVS Simulation. Simulation: Practice and Theory, vol. 5, (1997) 253–268
7. Kim, K.H., Kim, T.G., and Park, K.H.: Hierarchical Partitioning Algorithm for Optimistic Distributed DEVS Simulation of DEVS Models. Journal of Systems Architecture, vol. 44, (1998) 433–455
8. Object Management Group: The Common Object Request Broker: Architecture and Specification, 2.2 ed., Feb. (1998)
9. Message Passing Interface Forum: MPI-2: Extensions to the Message-Passing Interface, http://www-unix.mcs.anl.gov/mpi (1997)

The Effects of Network Topology on Epidemic Algorithms

Jesús Acosta-Elías[1], Ulises Pineda[1], Jose Martin Luna-Rivera[1], Enrique Stevens-Navarro[1], Isaac Campos-Canton[1], and Leandro Navarro-Moldes[2]

[1] Facultad de Ciencias, Universidad Autonoma de San Luis Potosí, San Luis Potosí, S.L.P., 78290, México, Av. Salvador Nava s/n, Zona Universitaria.
Tel: +52 (444) 826 2316, Fax: +52(444) 826 2321
{jacosta, mlr,u_pineda,icampos}@galia.fc.uaslp.mx,
http://www.fc.uaslp.mx

[2] Universitat Politécnica de Catalunya, Jordi Girona, 1–3, campus Nord, Barcelona, Spain
leandro@ac.upc.es

Abstract. Epidemic algorithms can propagate information in a large scale network, that changes arbitrarily, in a self-organizing way. This type of spreading process allows rapid dissemination of information to all network nodes. However, the dynamics of epidemic algorithms can be strongly influenced by the network topology. In this paper, numerical simulations are used to illustrate such influences. We address networks with simple topologies for simplicity and in order to isolate other effects that occur in more complex networks.

1 Introduction

Modern society increasingly relies on large scale computer and communication networks, such as Internet. A major challenge in this type of networks is the development of reliable algorithms for the dissemination of information from a given source (node) to thousands or millions of users (rest of the nodes in the network).

Epidemic data dissemination algorithms in computer and communication networks represent a mechanism analogous to the disease propagation in populations and the spread of rumor in social networks [14,15]. Therefore, these techniques potentially find a large spectrum of application such as mobile communication networks, P2P [17,16], and grid computing [2]. However, the development of new epidemic algorithms [12,7,5] for large scale network applications requires of a validation stage in order to analyze the behavior of important parameters in a network, *i.e.* efficiency, reliability, etc. Without a validation stage the system conditions can represent a complete mismatch between a predicted performance of the network and its real behavior.

In this paper, we are interesting to explore some properties of Networks that can influence the spreading process of epidemic algorithms. For example,

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 177–184, 2004.
© Springer-Verlag Berlin Heidelberg 2004

How much does the epidemic algorithm performance depend on the Network Topology?. To answer this question would allow us to separate any sort of improvement or loss introduced by the spreading process of an epidemic algorithm or the network topology.

Previous research works in this area are mainly focused in the characteristics and properties of epidemic algorithms, and its improvements, implemented in different types of network topologies. However, the effects of network topology on epidemic algorithms have received very little attention. For example, [18] proposes a new architecture that extends existing replication mechanisms to explicitly address only scalability and autonomy but without focousing in the topology influence issue. [18] automatically builds a logical update flooding topology over which replicas propagate updates to other replicas in their group. Replicas in a group estimate the underlying physical topology. Using this estimated network topology, a logical updated topology is connected for resilience. However, this work does not consider the effects of the Network topology on the performance of the epidemic algorithm. Also in [12] an analysis is done about the behavior of spreading computer viruses in scale-free networks, with particular emphasis in the propagation of virus. Different to [12], we do not investigate the topology influence from the viewpoint of combated spreading algorithms (viruses) instead we focus on non-combated spreading algorithm (epidemic).

In this paper, as a first step we investigate this problem by simulating the impact of simple network topology on the spreading of epidemic algorithms. This work seeks the behavior of epidemic algorithms applied on simple topological network structures such as a ring, line and mesh topologies, this will allow a better understanding of the results exposed here. In order to determine the degree of influence by the network topology, we make use of a weak consistency epidemic algorithm [3]. Replication with weak consistency is not new, it has already been applied as a mechanism of performance improvement and availability [11]. Weak consistency algorithms [3,1,6,13,10,9,4,11] consist in selecting a neighbor node to start updating a session. In a session two nodes mutually update their contents and thus, at the end of the session both nodes will have the same content. This process is called anti-entropy session [3] because the total entropy for each session in the network is reduced. For the rest of this paper we will be referred to this simply as a *session*.

The paper is organized as follows. First, the system model is presented in Section 2. A description of the experiments carried out for the various topology structures are shown in Section 3. Finally, results and conclusions are presented in Sections 4 and 5 respectively.

2 System Model

The model of our distributed system consists of a group of nodes $N = \{N_1, N_1, ... , N_n\}$ which communicate only by exchanging messages as shown in Fig. 1. We assume asynchronous (no bound on transmission delays) and reliable communication (a message sent by N_i to N_j is eventually received by N_j). Sites can

only fail by crashing (*i.e.* excluding Byzantine failures) but always can recover from this crash. It is assumed a fully replicated system, *i.e.* all nodes must have exactly the same content (every node contain copies of the same objects). Every node will be considered to be a server that gives services to a number of local clients (see Fig. 1). Clients make requests to a server, and every request is a *read* operation, a *write* operation or both. When a client invokes a *write* operation in a server, this operation (change) must be propagated to all servers (replicas) in order to guarantee the consistency of the replicas. An update is a message that carries a *write* operation to the replica in other neighboring nodes.

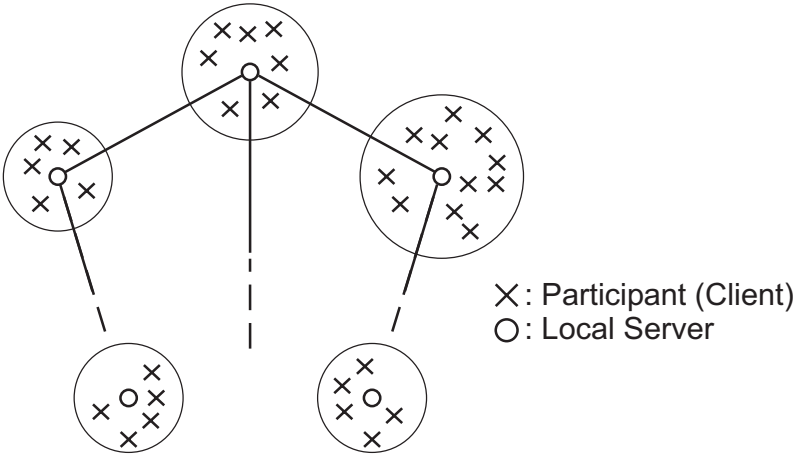


Fig. 1. General case of the considered model.

A weak consistent algorithm is used to maintain the consistency among the replicas. The algorithm has the following data structure:

- Timestamps [19] are used to provide an ordering upon events within the system. Timestamps are compared based only on their clock samples, so that they can be compared from different hosts. Timestamps are organized into time stamp vectors. A time stamp vector is a set of timestamps, each from a different node, indexed by their host. It represents a snapshot of the communication state in the system.
- The messages received for every node are kept in message log.
- Each node maintains a summary time stamp vector to record the messages updating process.
- Each node also maintains information about message acknowledgment from the rest of the nodes in the system.

Finally, we briefly describe the weak consistency algorithm used in this work. In our system we make use of a time stamp anti-entropy weak consistency algorithm [3]. This type of algorithm keeps the information from the time stamp

vectors and message log at each node in the system. This process is accomplished by exchanging periodically messages between a pair of nodes. From time to time, a node N_i selects a neighbor node N_j to start a session. A session between two nodes begins once a session request is accepted from any of the nodes. The next step is to exchange their summary time stamp and message acknowledgment vectors. What follows is that each node will determine if there are messages that the other node has not received yet, this is done by checking its summary time stamp. If any element in summary time stamp vector is greater than the equivalent element in the summary time stamp from the other node then the message or group of messages are obtained from its message log and sent to the other node. If a failure exists in the message exchange process then any node can abort the session and the changes made to the state of the nodes is discarded. A session ends with an exchange of the message acknowledgment. At the end of a successful session, both nodes have received the same set of messages.

3 Experiments

Some experiments have been carried out to study the system model considered here. A weak consistency algorithm simulator has been built using Network Simulator 2 [8]. For this, three different types of topologies, ring, line and mesh, are fed to the simulator. For each topology the experiment is carried out 5000 times with a confidence index equal to 99%. It is assumed that each node contains a new message at time equal to zero, that is, the system is in a non-consistent state. The Network reaches a consistent state once every node in the Network gather all messages sent from the rest of the nodes. At time equal to zero, the system is also said to be in a maximum stress because all nodes in the network hold a non-consistent state. Therefore, the aim is then to obtain a measure (level of consistency) related to the time spent (number of *sessions*) for each node to reach a consistent state. It can be interpreted as follows: the higher the number of nodes that reach the consistent state, the lower the entropy value. Once the system is in a consistent state the entropy is zero. The cumulative consistency state in the system can be expressed as follow

$$C(t) = \sum_{i=0}^{N} c(n_i, t) \tag{1}$$

with $c(n_i, t)$ defined as

$$c(n_i, t) = \begin{cases} 1 \text{ consistent state} \\ 0 \text{ non consistent state} \end{cases} \tag{2}$$

and where t is the session counter, n_i the ith node in the system and N the total number of nodes in the system.

Firstly, Fig. 2 illustrates the dissemination of information on a ring and line topologies. In the same way, Fig. 3 shows the spreading process of the epidemic algorithm on the mesh topology, in this case a Network with $N = 289$ nodes is considered where the nodes are distributed as a matrix of 17×17 nodes.

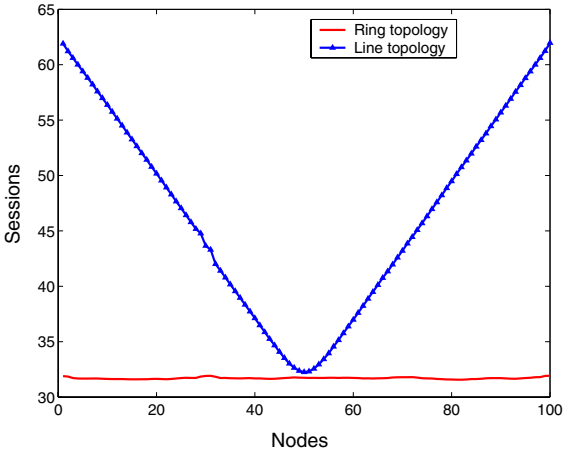


Fig. 2. Topology effects in the dissemination of information using epidemic algorithms.

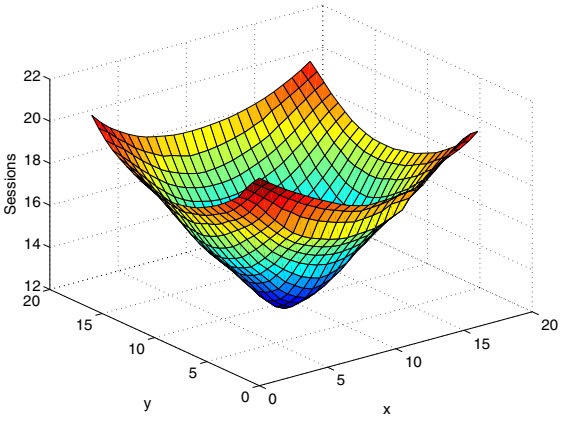


Fig. 3. Network performance using a mesh topology, 17×17 nodes.

4 Results and Discussions

In Fig. 2 is illustrated the results obtained when a ring and line topologies are considered. This Figure shows that using a ring topology all nodes reach the consistent state approximately at the same time ($\approx$ 32 *sessions* out of 100 nodes). In contrast, the nodes in a line topology reach the consistent state in a different number of *sessions*. For example, the nodes at the edges, see Fig. 2, take almost 62 *sessions* to receive all the messages, *i.e.* to reach the consistent state, while the central nodes take only 32 *sessions* to reach the same state.

Similar results are observed in the case of a mesh topology, see Fig. 3, as compare with the case of a line and ring topologies. That is, the nodes at the center take a number of *sessions* smaller than those nodes at the edge in order to reach the consistent state. It is also shown that a higher number of *sessions* is needed for the nodes that are gradually getting further away from the center of the mesh.

From the previous results, it is clear to see that the epidemic algorithm performance is influenced by the distance and connectivity patterns in the network, *i.e.* the network topology. As another example, we can see that in the line topology each node sees a different diameter with regard to other nodes in the network. It represents that each new message will be arriving to each node in the network with a different number of *sessions*. Here, the diameter is defined as the number of links required for a particular node to reach its furtherest node in the network. Therefore, in the line topology every new message will arrive sooner to the center nodes than to the edge nodes simply because the central nodes see a smaller diameter. For the ring topology case, this situation does not happen as all nodes sees the same diameter with regard to any other node in the Network. This behavior is clearly illustrated in Fig. 4 where we plot the diameter that every node sees in a Network with a line topology. This topology is implemented for the case of 10 nodes only. From Fig 4 it is shown that, for example, the node 0 sees a diameter of 9 while the node 4 sees only a diameter of 5, as it is expected.

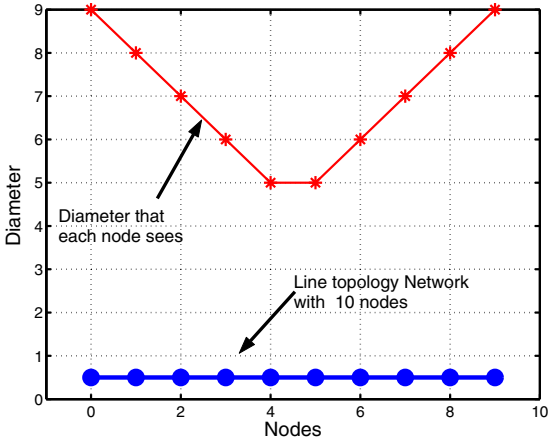


Fig. 4. Comparison of diameters from each node in a line topology Network of 10 nodes.

5 Conclusions and Future Work

In this paper the influence of network topology on epidemic data dissemination algorithms has been investigated. For this work, a weak consistency algorithm was chosen due to its wide use in the development of applications for large scale systems. Although the topologies considered in this paper are basically trivial, the structure of the performance functions can be quite general and is not completely known.

A number of experiments were carried out with the help of numerical simulations. For this purpose, a simulator was developed using network simulator 2. The network topologies investigated were a line, ring and mesh type. From the simulation results, it is evident that a significant effect exist from the network topology on the performance of this sort of epidemic algorithms. For example, the nodes that experience a diameter distance greater than the rest (case of line topology) require a greater number of sessions to reach a consistent state. In contrast, the ring topology requires almost the same number of sessions for all nodes. Finally, for future work it would be interesting to evaluate and analyze such properties on more complex topologies.

References

1. Atul Adya: Weak Consistency: A Generalized Theory and Optimistic Implementations for Distributed Transactions, PhD thesis M. I. T., Department of Electrical Engineering and Computer Science, March 1999.
2. I. Foster, C. Kesselman, S. Tuecke.The Anatomy of the Grid: Enabling Scalable Virtual Organizations International J. Supercomputer Applications, 15(3), 2001.
3. R. A. Golding, "Weak-Consistency Group Communication and Membership", PhD thesis, University of California, Santa Cruz, Computer and Information Sciences Technical Report UCSC-CRL-92-52, December 1992.
4. R. Guy, J. Heidemann, W. Mak, T. Page, Jr., G. Popek and D. Rothmeier. Implementation of the Ficus Replicated File System. Proceedings Summer USENIX Conf. June 1990.
5. JoAnne Holliday, Divyakant Agrawal, Amr El Abbadi, "Partial Database Replication using Epidemic Communication", Proceedings of the 22 nd International Conference on Distributed Computing Systems (ICDCS2002).
6. Brian Kantor, Phil Lapsley RFC0977, http://www.ietf.org/rfc/rfc0977.txt, 1986
7. L. Li, J. halpern, Z. J. Hass: Gossip-based ad hoc routing, Proceedings of Infocom, 2002, pp. 1707-1716.
8. The Network Simulator: http://www.isi.edu/nsnam/ns/
9. K. Petersen, M. J. Spreitzer, D. B. Terry, M. M. Theimer, and Demers: Flexible Update Propagation for Weakly Consistent Replication, Proceedings of the 16th ACM Symposium on Operating Systems Principles (SOSP-16), Saint Malo, France, October 5-8, 1997, pages 288-301.
10. M. Satyanarayanan, Scalable, Secure, and Highly Available Distributed File IEEE Computer May 1990, Vol. 23, No. 5
11. Michel D. Schroeder, Andrew D. Birrel, and Roger M. Needham, Experience with Grapevine: The Growth of a Distributed System, ACM Transactions of Computer Systems, Vol. 2, No. 1, February 1984, Pages 3-23.

12. R. Pastor-Satorras and A. Vespignant. "Epidemic Spreading in Scale Free Networks", Physica Review Letters, 86, 2001.
13. Walter Willinger, Ramesh Govindan, Sugih Jamin, Vern Paxson, and Scott Shenker Scaling phenomena in the Internet: Critically examining criticality. PNAS, February 19, 2002, vol. 99, suppl. p. 1 2573-2580.
14. I. Pool and M. Kochen. Contacts and influence. Social Networks, 1:5(51, 1978).
15. Fang Wu, Bernardo A. Huberman, Lada A. Adamic, and Joshua R. Tyler: Information Flow in Social Groups, annual CNLS conference on Networks, Santa Fe, NM, May 12, 2003.http://www.hpl.hp.com/shl/papers/flow/
16. Francisco Matias Cuenca-Acuna, Christopher Peery, Richard P. Martin, Thu D. Nguyen:PlanetP: Using Gossiping to Build Content Addressable Peer-to-Peer Information Sharing Communities, In proc. of the IEEE International Symposium on High Performance Distributed Computing (HPCD 12), Seattle, WA, June 2003
17. A. Ganesh and A. Kermarrec and L. Massoulie:Peer-to-Peer Membership Management for Gossip-based Protocols. IEEE Trans. Comp., 52(2):139–149, Feb. 2003.
18. Katia Obraczka: Massively Replicating Services in Wide Area Internetworks, PhD dissertation, Computer Science Department, University of Southern California, December 1994.
19. Leslie Lamport. Time, clocks, and the ordering of events in a distributed system. Communications of the ACM, 21(7):558(1978).

A Systematic Database Summary Generation Using the Distributed Query Discovery System

Tae W. Ryu[1] and Christoph F. Eick[2]

[1]Department of Computer Science, California State University, Fullerton
Fullerton, CA 92834, USA
tryu@fullerton.edu
[2]Department of Computer Science, University of Houston, Houston
Houston, TX 77204, USA

Abstract. This paper introduces an approach to generate a database summary systematically using the distributed query discovery system, MASSON. Our approach is first to create an object-view and partition the database based on the object-view into clusters with similar properties, and then to generate the summary for each cluster. For this purpose, we propose a data set representation framework and introduce a proper similarity measure framework. The paper also describes the techniques used to generalize the generated primitive summary descriptions by MASSON and to improve the performance of the system using clustered computers and CORBA.

1 Introduction

For the current flood of data routinely generated by many organizations/Web sites, automatic tools and techniques have become necessary for analyzing the data. Recently many techniques, tools, and systems designed to discover interesting knowledge from the huge amount of data have been proposed/developed or are still under development. Some popularly used methods include classification, clustering, regression, data summarization, dependency modeling, link analysis, change and deviation detection, visualization, etc. [3].

This paper centers on the problem of database summary generation. Database summarization is useful in comprehending a database because it may provide a compact representation of the core properties of the database. It may also help us make useful inferences from a database. For example, a simple statement "Americans do not like high-fat food," which can be a summary of a market database, allows a food company to infer advertising a new food product from the company as a low-fat product. However, generating a good summary for a database is a non-trivial problem because we need to deal with the structural data sets (e.g., typical relational or object-oriented database consists of many related relations or classes.) Several approaches to the database summary generation have been proposed. Lee and Kim [13] propose hypotheses refinement approach using fuzzy logic to summarize a database and produce a set of high-level abstract terms. An integrated system, EXPLORA [10],

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 185–195, 2004.
© Springer-Verlag Berlin Heidelberg 2004

searches interesting summary for a data set in the form of statement types using application-specific domain knowledge and statistical knowledge. Dhar and Tuzhilin [4] discuss user specified abstract function and aggregate principles to provide an abstract form of summary. Other approaches include generating rules for associative information between attributes [1] or for discriminating descriptions for the given data set [14].

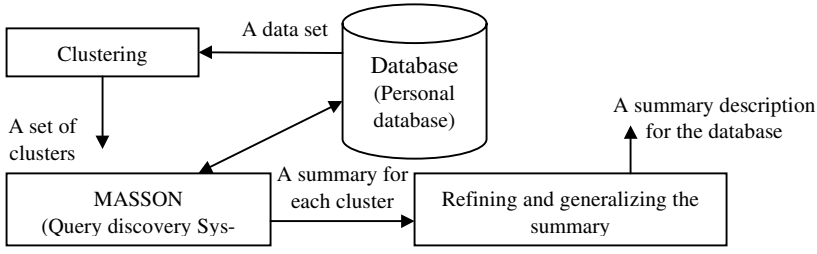


Fig. 1. Steps for database summary generation

Our approach is to partition the given database into clusters and generate the summary descriptions for each cluster. Fig. 1 shows the major steps involved in generating a database summary in this research. First, the data set to be analyzed is selected from the given database and is preprocessed. Second, the preprocessed data set is partitioned into a set of clusters with similar properties. Third, a summary is generated for each cluster. The generated summary can be further refined and generalized to produce the high-level summary descriptions for the database. For this purpose, we introduce a data set representation framework and a proper similarity measure framework for the object-based clustering. For the summary generation for each cluster, we use MASSON, a database query discovery system [18] using genetic programming (GP) [12]. In addition, we introduce the distributed MASSON developed based on clustered computers and CORBA for the performance improvement.

2 Clustering Structured Databases

The flat file is the simplest and most frequently used format in the traditional data analysis area. When using flat file format, data objects are represented through vectors in *n*-dimensional attribute space, each of which describes an object, and the object is characterized by *n* attributes, each of which has a single value. Most existing data analysis and data mining approaches assume that data sets to be analyzed are represented in a flat file format. Due to the fact that databases are more complex than flat files, database clustering faces following additional problems that do not exist when clustering flat files:

- **Support for object-view**: Databases contain objects that belong to different types; consequently, it has to support an object-view that consists of a subset of objects in the database. It is not reasonable to apply a clustering algorithm directly to the database without focusing on a view of the database. To illustrate this

problem, let us assume we have a relational database that consists of several related relations, e.g., person, bank, transaction, purchase, etc. We need to determine which relation should be the focus for clustering.

- **Problems with relationships**: Databases contain *1:1*, *1:n* and *n:m* relationships between objects of the same and different types. Therefore, an object in the object-view defined must include the *1:n* or *n:m* related information. For example, consider two relations, person and purchase with *1:n* relationships (a person can make many purchases). It is important to decide how this type of related information should be represented in the input data set for a selected clustering algorithm.
- **Object similarity measures**: The definition of object similarity is more complex due to the presence of related information that characterizes an object. So, it may have to define different similarity measures for the objects in the object-view.

Our approach for each of these problems will be discussed in more detail as follows.

2.1 A Framework for Data Set Representation

The data set representation for clustering can be considered either object-based or record-based. The main distinction between these two approaches is that in object-based clustering the data unit for clustering algorithm is an object that consists of all the related information as well as its properties [6], [19] whereas in the conventional record-based clustering the data unit is a record that consists of only properties and the related information is represented in different records. In this research, our goal is to generate summary descriptions for each group of objects with similar properties. So, we partition the selected data set into groups using the object-based clustering approach. Note that the record-based clustering is not appropriate here since one object can be represented in many different records in the record-based data set when there is a *1:n* relationship between two classes/relations.

In general, our framework consists of the following mechanisms:

- An object identification mechanism that defines what classes of objects will be clustered and how those objects will be uniquely identified,
- Mechanisms to define modular units based on object similarity have to be provided; each modular unit represents a particular perspective of the objects to be clustered; similarity of different modular units is measured independently. In the context of the relational data model modular units are defined as procedures that associate a bag of tuples with a given object. Using this framework, objects to be clustered are characterized by a set of bags of tuples, one bag for each modular unit.

To illustrate this framework, let us assume we are interested in clustering a database *Persons*. In this case the attribute *pid* of the relation *person* that uniquely identifies persons serves as our object identification mechanism. After the object identification mechanism has been selected, the relevant attributes to define similarity between persons need to be selected. In the particular case, we assume we consider the person's age/gender information, the amount of money they spend on various product groups, and the person's daily spending pattern to be relevant for defining person similarity. In the next step, modular units to measure person similarity have to be defined. In this particular example, we identify three modular units each of which characterizes per-

sons through a set of tuples. For example, the person with *pid* 1 is characterized as a 43 years old male, who spent 400, 70, and 200 dollars on product groups p1, p2, and p3, and who purchased all his goods in a single day of the reporting period, spending total 670 dollars. There are different approaches to define modular units. When the relational data model is used, modular units can be defined using SQL queries that associate persons (e.g., using *pid*) with a set of tuples that are specific for the modular unit. For example, the following 3 SQL queries associate persons with the characteristic knowledge with respect to each modular unit:

```
Modular Unit 1 := SELECT pid, age, gender
                  FROM Person;

Modular Unit 2 := SELECT person.pid, pgid, amount
                  FROM Person, Purchase
                  WHERE person.pid=Purchase.pid;

Modular Unit 3 := SELECT person.pid, sum(amount), purdate
                  FROM person, Purchase
                  WHERE person.pid=Purchase.pid
                  GROUPED BY person.pid, purdate;
```

Many different object views can be constructed from a given database. In the above example, a more complicated scheme for defining object views that characterizes objects through sets of bags of tuples, has been introduced. We claim that this data set representation framework is more suitable for database clustering.

2.2 A Framework of Similarity Measures for Object-Views

When defining object similarity in this framework we assume that a similarity measure is used to evaluate object similarity with respect to a particular modular unit. Object-similarity itself is measured as the weighted sum of the similarity of its modular units. More formally, let O be the set of objects to be clustered, $a, b \in O$ and $m_i: O \rightarrow X$ denotes a function that computes the bag of tuples of the i-th modular unit. Θ_i and w_i denote the similarity function and the weight (Higher weights are assigned to more important attributes) for the i-th modular unit. n is the number of modular units in an object. Based on these definitions, the similarity Θ between two objects a and b can be defined as follow:

$$\Theta(a, b) = \sum_{i=1}^{n} w_i \Theta_i(m_i(a), m_i(b)) / \sum_{i=1}^{n} w_i$$

In many real world problems, we often encounter a data set with a mixture of attribute types. Specifically, if algorithms are to be applied to databases, it may not be sensible to assume a single type of attributes since the data set can be generated from multiple relations with different properties in a given database. There are many similarity metrics and concepts proposed in the literature from variety of disciplines including engineering, science, and psychology [7]. Wilson and Martinez [22] introduce a comprehensive similarity measure called, HVDM, IVDM, and the WVDM for handling mixed types of attributes. We used the similarity measure framework proposed by Gower [9] to handle mixed types of attributes. In addition, for data type specific similarity measure, we employ the similarity measure by [21] for qualitative types and the modified Euclidean distance metric by [20] for quantitative types. Note that these

similarity metrics are group-similarity measures. Our approach to assess object similarity associates the similarity measures with modular units that represent different facets of objects.

We also use the contextual similarity measure implemented in [20]. Let us assume the similarity of attribute β has to be evaluated in the context of attribute α, which we denote by: β|α. Then we can define the similarity between two objects *a* and *b* having attributes α and β as the similarity of β attribute with respect to α attribute. The new similarity function is defined as follow:

$$s_{\beta|\alpha}(a,b) = \sum_{k} \delta_{\alpha}^{(k)} s_{\beta}^{(k)} / \sum_{k} \delta_{\alpha}^{(k)},$$

where δ_α is a matching function for the attribute α and s_β is a similarity function for the attribute β, *k* is number of elements in a *bag*. The value of δ_α is 1 for qualitative attribute if both objects take the same value for the attribute α, otherwise, δ_α is 0 (e.g., no matching values). The value of δ_α is between 0 and 1 for quantitative attribute (e.g., a normalized distance value). In this definition, the information from the related multi-valued attributes is combined in an orderly way to give a similarity value. This similarity measure is embedded into our similarity framework. Note that the proposed contextual similarity is not designed to find any sequential patterns like *PrefixSpan* [17] or to measure transitive similarity [8] but to take the valid contextual information into account of the similarity computation.

2.3 Clustering Algorithms

For the clustering algorithm that can handle the data set represented in the proposed framework, any algorithm such as *Nearest-neighbor*, *k-medoids*, etc. that computes the similarity measure directly between two objects can be used. In this research, we selected the *Nearest-neighbor* algorithm. However, we found that generalizing the algorithm, *K-means* is not trivial, because of difficulties in computing *centroids* for clusters of objects that are characterized by sets of bags of values.

3 Summary Generation Using Genetic Programming

3.1 Summary Generation Using MASSON

Ryu and Eick introduced a query discovery system, MASSON and proposed the *deriving queries by results* approach to generate the common characteristics information for a given set of objects [18]. In MASSON, if a set of objects is provided, the system generates many dynamic query structures for the given object set by accessing the database specified by the user for schema information. The generated queries are sent to the database for execution. The system will then evaluate those queries based on how well the returned results from the database cover the given object set. A query that perfectly matches the given object set is called *perfect hit* query. Those non-perfect hit query structures are then genetically evolved based on their fitness values using GP. The discovered query set can distinctively describe the commonalities for the given set of objects against other objects in the database. As the database query

language, MASSON uses the basic navigational query operators, *SELECT*, *RESTRICTED*, *RELATED*, *GET-RELATED*, and set operators. *SELECT* operator selects all the objects that satisfy the conditional predicate. *RESTRICTED* operator restricts the objects in the given set to those that are related to another class, according to the given predicate. The relationship operator *RELATED* selects all the objects from a class that are related to objects in another class through the relationship links. *GET-RELATED* operator is an inverse operator of *RELATED*. In addition, the set operators *UNION*, *INTERSECTION*, *DIFFERENCE* are supported. Following example queries illustrate some of discovered queries and the corresponding semantics for a cluster that consists of a set of objects.

q$_1$: *"Persons who have transferred cash to anyone more than 4 times and are age less than 30 and greater than 20."*

```
(GET-RELATED (RESTRICTED BANK-ACCOUNT (> TRANSFER-TO 4))
                    OWNED-BY
                (SELECT PERSON (AND (< AGE 30) (> AGE 20))))
```

q$_2$: *"Persons that have more than 7 times of suspect activities records or spent more than $5,000 cash in a store."*

```
(O-UNION (RELATED person shopped-at
                    (SELECT purchase (AND (> amount-spent 5000) (= payment-type 1))))
            (SELECT person (> nsuspect-act 7)))
```

For the database summary generation, we provide MASSON with each cluster (a set of objects) with similar properties and use database query as a summary representation language. Database query language can be effectively used for describing the summary of a cluster since the query expresses the given data set that it computes [18]. One major advantage of this approach is that the discovered queries can represent the structural information implicitly stored in related classes/relations in a database as well as other characteristic information.

3.2 Summary Generalization

Our navigational query operators can represent structural flow of relationships between classes during searching for interesting queries. However, interpreting those system-generated queries is rather difficult for the end user. We need an interpretation tool/query simplifier for the end user to help him/her better understand the queries returned by the system. We also need a query simplifier to remove redundant computations and to normalize the queries. One problem that the search process faced in the original MASSON was that the same query can be written in too many different ways, which slows down the search process. In this section we discuss how to simplify and generalize the queries to the user-understandable higher-level summary descriptions.

To illustrate this, we assume that an object-view created from the *Persons* database is clustered into k clusters, *Persons* = $\{g_1, g_2, \ldots, g_k\}$. Then we provide MASSON with $\{g_1, g_2, \ldots, g_k\}$ to generate a set of queries, $Q_i = \{q_{i1}, q_{i2}, \ldots, q_{in}\}$ for each cluster g_i, and n is the number of queries for g_i through the steps: $g_1 \rightarrow$ MASSON $\rightarrow Q_1 = \{q_{11}, q_{12}, \ldots, q_{1m}\}$, $g_2 \rightarrow$ MASSON $\rightarrow Q_2 = \{q_{21}, q_{22}, \ldots, q_{2o}\}$, …, $g_k \rightarrow$ MASSON $\rightarrow Q_k = \{q_{k1}, q_{k2}, \ldots, q_{kr}\}$. We assume each query $q_{ij} \in Q_i$, as a *perfect hit* query. By using the similar approach to [23], the set of queries in Q_i are simplified and generalized to

describe a summary for the cluster g_i. Predicate conditions on attribute x in two queries q_{i1} and $q_{i2} \in Q_i$ can be combined and generalized to new predicate conditions by the rules.

R1: $q_{i1}:(a \le x \le b),\ q_{i2}:(c \le x \le d) \rightarrow q_{i3}:(\min(a,c) \le x \le \max(b,d))$

R2: $q_{i1}:(x = a),\ q_{i2}:(x = b) \rightarrow q_{i3}:(x = a \text{ or } x = b)$

R3: $q_{i1}:(a \le x \le b),\ q_{i2}:(x = c) \rightarrow q_{i3}:(a \le x \le b) \text{ if } a \le c \le b$

The rules R1 and R2 are applied to range conditions and equality conditions, respectively. Range conditions can be generalized to an extended range condition. For example, the range conditions ($5 < x < 10$) and ($2 < x < 20$) can be generalized to a condition ($2 < x < 20$). For the predicate with equality condition, there can be two kinds of conditions: numeric and symbolic conditions. The numeric equality conditions can be generalized disjunctively by combining two conditions using OR operator. For example, the conditions ($x = 10$) and ($x = 200$) can be generalized to a condition ($x = 10$ or $x = 200$). The rule R3 is applied to the results of R1 and R2. For example, the conditions ($x = 10$ or $x = 200$) and ($x \ge 10$) can be generalized to a condition ($x \ge 10$). The equality condition with symbolic values (e.g., nominal values) can be generalized based on the class hierarchy in a database or predefined hierarchy definitions by the user. For example, if a class A is in higher level of a class hierarchy than the class B in a database, then the condition ($x = a$) is more general than the condition ($x = b$), where a, b are the values of the classes A, B, respectively. So, the condition ($x = b$) can be generalized to ($x = a$).

After the generalization process, all the queries q_{i1}, q_{i2}, …, q_{in} in a cluster g_i are combined to form the summary descriptions Dg_i for the cluster. Then we have the summary of a data set, *Persons* by collecting all the descriptions Dg_1, Dg_2, …, Dg_k. As a more specific example, we can have the following generalized queries from queries Q_1, Q_2, and Q_3 as follows:

Q_1: (SELECT *Person* (> *age* 20))
(SELECT *Person* (> *age* 30)) → (SELECT *Person* (> *age* 20))

Q_2: (RELATED *Person has* (RESTRICTED *Phone* (> *made-call* 20)))

Q_3: (GET-RELATED (SELECT *Bank-Account* (AND (> *balance* 1500) (< *balance* 2800))) *owned-by Person*)
(GET-RELATED (SELECT *Bank-Account* (> *balance* 1500)) *owned-by Person*)
→
(GET-RELATED (SELECT *Bank-Account* (> *balance* 1500)) *owned-by Person*)

These queries can be interpreted as a linguistic summary description: *"The persons who are age over 20, have bank balances more than $1,500, and make phone calls more than 20 times."* As shown in this example, we can possibly transform the primitive queries generated by MASSON to the user-understandable format of higher-level summary description. We used the same database *Persons* used in [18] (that consists of personal activities) for experiment. From this experiment, we have 11 clusters. MASSON initially generated more than 10 queries for each cluster. These queries were later simplified/generalized to less than 5 queries for each cluster.

3.3 Performance Improvement in Distributed Computing Environment

Like many GP-based systems or other data mining systems, performance is the major problem in MASSON [11], [15]. Particularly, in MASSON, many queries (e.g., 100 queries, depending on the size of population) are generated and executed for each generation. Each query requires database access, which takes much longer time than an individual running within main memory in typical GP-based systems. For the performance improvement, we built a distributed computing environment with clustered computers using the Common Object Request Broker Architecture (CORBA) [16].

Among several parallelism techniques for databases, we employ interquery parallelism approach to improve the performance of MASSON since our application has two useful properties for parallelization. First, all the queries generated by the system are read-only queries. Therefore, no locking and logging is needed. We don't have to worry about the dependency among queries. Second, the database can be replicated in several machines. The performance may increase almost linearly with the degree of parallelism and replication. For our parallel architecture, we employ clustered-computers approach, which is one of the popular and economical parallel architectures available. The rapid growth of interconnected high performance PCs (or workstations) has produced a new computing paradigm called Cluster of Workstations (COW). The use of COW provides several benefits [2]: first, clusters provide better performance by using a number of workstations at the same time. Second, heterogeneous computers can be worked together harmoniously and each computer can function in autonomous way. Third, clusters provide incremental scalability of hardware resources since additional workstation can be added and easily scaled into large configurations.

To implement the distributed MASSON, we use the CORBA based on the Distributed Computing Environment (DCE). The DCE is from the Open Software Foundation (OSF), which provides the tools and framework to build distributed applications on top of a multi-vendor environment. CORBA allows computing objects to be distributed across a heterogeneous network so that the distributed object components are interoperable as a unified system regardless of their platforms or environments. For implementation, the Java and a free CORBA middleware, Jonathan [5] is used. In the distributed MASSON, there is one master computer that manages the overall query discovery process and several slave computers that perform the query execution and evaluation. A database is replicated in each slave computer. The master computer generates queries and distributes them to slave computers. Each slave computer receives the distributed queries, executes them in local database, and returns the evaluated result of each query back to the master computer. The master computer receives the result from each slave and selects queries based on the probability of selection. If a query that made the *perfect hit* is found in a slave computer, the query is stored in a shared disk by the slave computer. The master computer checks the shared disk to collect the perfect hit queries from each slave computer. The main evolutionary process (e.g., generating a new population for the next generation) is performed in the master computer. Another major task of the master computer is a simple load balancing. The master computer checks the current progress of each computer and bal-

ances the load of the whole system by redistributing the queries to be executed to other less busy computers.

It is not difficult to see the performance improvement of MASSON in the distributed computing environment. Let T be the total time taken to finish a generation in the original MASSON, the time t taken to finish one generation in the distributed MASSON can be $t = T/n + \alpha$, where n is the number of clustered computers, α is the overhead such as communication time and other time taken for controlling the overall process. The overhead can be defined as, $\alpha = \mu * n + \omega$, where μ is the communication/control time and ω is the initial setup time.

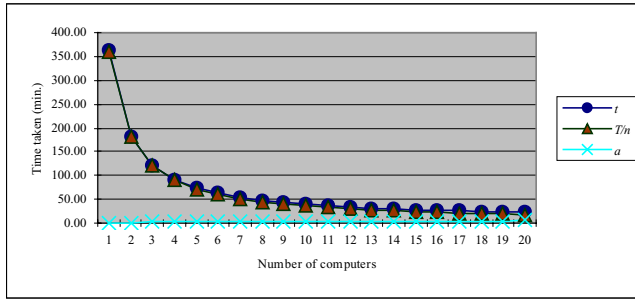


Fig. 2. Performance improvement of distributed MASSON

Fig. 2 shows the performance improvement in the distributed MASSON. The brief specification of the computers used in this experiment is Pentium 800MHz with 125 MB RAM installed in a Local Area Network (LAN). The time taken to finish the job with one computer was about 6 hours (370 minutes). The time taken in the distributed environment was shortened to about 25 minutes with 20 computers. As shown in Fig. 2, the performance improvement is not linear mainly because of the additional overhead, α. Moreover, accessing additional secondary storages by adding more computers requires additional overhead, which might limit the scalability of the system.

4 Conclusion

Database summary is a challenging problem but very useful in understanding large amounts of structured data. In this paper, we proposed a systematic database summary generation approach using the distributed query discovery system, MASSON, implemented with clustered computers and CORBA. In this system, we used an object-based data set representation and group similarity measure framework for structured database clustering. The proposed framework can cluster a set of objects with related information using the modular units and group similarity measures. We claim that the proposed object-based data representation and similarity measure framework is more appropriate than the conventional record-based representation in clustering a set of objects in a structured database. The query language for the intermediate summary representation is effective enough to generate higher-level summary descriptions through the proposed generalization approach.

References

1. Agrawal, R. Imielinski, T., and Swami, A.: Mining Association Rules between Sets of Items in Large Databases. *Proc. ACM SIGMOD* (1993) 207-216
2. Anderson, E., Culler, D. and Paterson, D.: A Case of NOW (Network of Workstations). IEEE Micro, 15(1): 54-64 (1995)
3. Chen, M-S., Han, J., and Yu, P.S.: Data mining: An Overview from a database perspective. IEEE Transactions on knowledge and data engineering, Vol. 8, No. 6 (1996)
4. Dhar, V. and Tuzhilin, A.: Abstract-Driven Pattern Discovery in Databases. IEEE Transactions on Knowledge and Data engineering. Volume 5 (1993)
5. Dumant, B., Tran, D., Horn, F., and Stefani, J-B.: Jonathan: an Open Distributed Processing Environment in Java, Middleware'98: IFIP International Conference on Distributed Systems and Open Distributed Processing, The Lake District, UK (1998)
6. DuMouchel, W., Volinsky, C., Johnson, T., Cortes, C., and Pregibon, D.: Squashing Flat Files Flatter. Proc. Of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-99), San Diego, California, USA (1999)
7. Everitt, B.S.: Cluster Analysis. Edward Arnold, Copublished by Halsted Press and imprint of John Wiley & Sons Inc., 3rd edition (1993)
8. Gibson, D., Kleinberg, J., and Raghavan, P.: Clustering Categorical Data: An Approach Based on Dynamical Systems. Proc. of the 24th International Conference on Very Large Databases, New York, USA (1998)
9. Gower, J.C.: A general coefficient of similarity and some of its properties. Biometrics 27, (1971) 857-872
10. Hoschka, P. and Klösgen, W.: A Support System for Interpreting Statistical Data. Knowledge Discovery in Databases. MIT Press, Cambridge, MA. (1991).
11. Kimm, H.L. and Ryu, T.W.: A Framework for Distributed Knowledge Discovery System over Heterogeneous Networks using CORBA, Proc. of the ACM SIGKDD-00 Workshop on Distributed and Parallel Knowledge Discovery, Boston, Massachusetts (2000)
12. Koza, J.R.: Genetic Programming: On the Programming of Computers by Means of Natural Selection. Cambridge, MA: The MIT Press (1990)
13. Lee, D.H. and Kim, M.H.: Discovering Database Summaries through Refinements of Fuzzy Hypotheses. IEEE Transactions on Knowledge and Data engineering, Volume 5 (1993)
14. Mitra, S. and Hayashi, Y.: Neuro-Fuzzy Rule Generation: Survey in Soft Computing Framework. IEEE Transactions on Neural Networks, Vol.11, No.3 (2000) 748-768
15. Neri, F. and Giordana, A.: A parallel genetic algorithm for concept learning. Proc. 6th International Conference on Genetic Algorithms, (1995) 436-443
16. Otte, R., Patrick, P. and Roy, M.: Understanding CORBA: The common object request broker architecture, Prentice Hall (1996)
17. Pei, J., Han, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U., and Hsu, M-C.: Prefix-Span: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth. Proc. of the 17th International Conference on Data Engineering, Heidelberg, Germany (2001)
18. Ryu, T.W and Eick, C.F.: Deriving Queries from Results using Genetic Programming. Proc. of the 2nd Int'l Conf. on Knowledge Discovery and Data Mining, Portland, Oregon (1996)
19. Ryu, T.W. and Eick, C.F.: Similarity Measures for Multi-valued Attributes for Database Clustering. Proc. of the International Conference on SMART ENGINEERING SYSTEM DESIGN (ANNIE'98), St. Louis, Missouri (1998)

20. Ryu, T.W., Chung, H., Chang, W., and Salameh, H.: Database Clustering vs. Flat File Data Clustering. Proc. of the International Conference on Artificial Intelligence, Las Vegas (2001)
21. Tversky, A.: Feature of Similarity. Psychological review, 84(4): (1977) 327-352
22. Wilson, D.R. and Martinez, T.R.: Improved Heterogeneous Distance Functions. Journal of Artificial Intelligence Research, 6 (1997) 1-34.
23. Zhong, N. and Ohsuga, S.: Managing/refining structural characteristics discovered from databases. Proc. of the 24th Hawaii International Conference on System Sciences, Volume 3, (1995) 283-292

Parallel Montgomery Multiplication and Squaring over GF(2^m) Based on Cellular Automata

Kyo Min Ku [1], Kyeoung Ju Ha [2], Wi Hyun Yoo [3] , and Kee Young Yoo [4]

[1] Mobilab Co. Ltd, Plus B/D 4F 952-3, Dongcheon-dong, Buk-gu, Daegu, Korea, 702-250
kmku@mobilab.co.kr
[2] Daegu Haany University, 75 San, JumChon-Dong, Kyungsan, KyungPook, Korea, 712-715
kjha@kyungsan.ac.kr
[3] Samsung Electronics Co. Ltd, 94-1, Imsoo-Dong, Gumi-City, KyungPook, Korea, 730-350
whyoo4003@samsung.com
[4] Kyungbuk National University, 1370 Sanguk-dong, Buk-gu, Daegu, Korea, 702-701
yook@knu.ac.kr

Abstract. Exponentiation in the Galois Field GF(2^m) is a primary operation for public key cryptography, such as the Diffie-Hellman key exchange, ElGamal. The current paper presents a new architecture that can simultaneously process modular multiplication and squaring using the Montgomery algorithm over GF(2^m) in m clock cycles based on a cellular automata. The proposed architecture makes use of common-multiplicand multiplication in LSB-first modular exponentiation over GF(2^m). In addition, modular exponentiation, division, and inversion architecture can also be implemented, and since cellular automata architecture is simple, regular, modular, and cascadable, it can be utilized efficiently for the implementation of VLSI.

1 Introduction

Galois fields have numerous practical applications in modern communication systems, including error-correcting codes, switching theory, cryptography, and digital signal processing[1]. The current authors are particularly interested in cryptographic applications where m is very large. One example of a cryptographic application is the Diffie-Hellman key exchange algorithm[2]. This system is based on discrete exponentiation and reduction modulo $n(x)$ over the field GF(2^m). Discrete exponentiation can also be used to perform public-key data exchange and digital signatures, as shown by ElGamal[3]. The exponentiation operation can be implemented using a series of squaring and multiplication operations in GF(2^m) using the binary method. Also, the Elliptic Curve Cryptosystem is based on constant multiplication[4]. The algorithms used to implement the multiplier include the LSB-first multiplication algorithm[5], MSB-first multiplication algorithm[6], and Montgomery algorithm[7].

Most previous research activities on Montgomery multiplication/squaring have focused on a systolic array, and one of the latest developments is that Montgomery multiplication can be performed in $3m$ clock cycles using m+1 cells[10].

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 196–205, 2004.
© Springer-Verlag Berlin Heidelberg 2004

The current study is the first time Montgomery multiplication/squaring has been attempted using a CA. Cellular automata(CA) architecture is simple, regular, modular, and cascadable, as such it can be utilized efficiently for the implementation of VLSI and is currently applied in many areas, including the encryption, decryption, etc. of a crypto-system[8][9].

The current paper proposes a new structure in which modular multiplication and squaring using the Montgomery algorithm are processed at the same time for effective exponentiation on GF(2^m) using a CA.

Accordingly, a structure is designed that can accomplish modular multiplication and modular squaring simultaneously using a CA. As a result, Montgomery multiplication /squaring can be performed in m clock cycles using m cells, 3m-1 AND gates, 3m-1 XOR gates, 1 NOT gate, and 5 registers using a CA, which is much more efficient from the perspective of time and space than [10]. This is facilitated by identifying the parts of the modular multiplication and squaring computation that can be performed in common and then processing the remainder in parallel.

The outline for the rest of this paper is as follows. Chapter 2 gives an overview of a cellular automata. Chapter 3 describes the modular exponentiation algorithm for Montgomery multiplication over GF(2^m). Chapter 4 presents a structure for performing both modular multiplication and squaring using a CA. Finally, chapter 5 analyzes the performance and gives some conclusions.

2 Cellular Automata (CA)

Cellular automata consist of numbers of interconnected cells arranged spatially in a regular manner[8][9]. The next state of a cell depends on the present states of 'k' of its neighbors, for a k-neighborhood CA. Example of one rule of a 2-state 3-neighbor 1-dimensional CA is shown below.

State of neighbor	111	110	101	100	011	010	001	000
Next state	0	1	0	1	1	0	1	0

(Rule 90)

In this case, the state of the neighbors refers to 8 available states of 3 neighbors at time t. Among the 3 bits used to indicate the states, the middle bit represents the state of the cell itself, while the left and right bits indicate the states of the left and right neighbors, respectively. Rule 90 shows the state of the ith cell at time t+1, where 90 means the 8 bits of the next state shown in a decimal system.

The present state of a CA with n cells can be shown in terms of n-vector $x=(x_0x_1\ldots x_{n-1})$, where x_i is the value of Cell i, and x_i is an element of GF(2). The next state of a linear CA can be determined by multiplying the characteristic matrix with the vector in the present state, where the characteristic matrix shows the entire rules of the CA. Assuming that x^t is the state of the CA at time t (regarded to be one row vector) and the characteristic matrix is B, the state of the CA at time t+1 can be expressed as follows:

$$x^{t+1} = B\,x^t$$

where the arithmetic computation occurs on GF(2).

The characteristic matrix under rule 90 is as follows:

$$\begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 & 0 \\ 1 & 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & 1 & \dots & 0 & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & 1 & 0 \end{bmatrix}$$

In the above example, it is shown that element '1' of the matrix on the *i*th line of the *j*th row shows that the *i*th cell is dependent on the neighbor of the *j*th cell.

3 Montgomery Multiplication and Modular Exponentiation Algorithm

3.1 Montgomery Multiplication Algorithm

This chapter presents general algorithms for Montgomery multiplication over GF(2^m)[11]. Instead of computing $a(x)\cdot b(x)$ over GF(2^m), the Montgomery Algorithm computes $a(x)\cdot b(x)\cdot r(x)^{-1}$ over GF(2^m), where $r(x)$ is a special fixed element of GF(2^m). A similar idea was proposed by Montgomery in [7] for the modular multiplication of integers. Montgomery's technique is also applicable to the field GF(2^m)[11]. The Montgomery multiplication method requires that $r(x)$ and $n(x)$ are relatively prime, i.e., $gcd(r(x), n(x))=1$. For this assumption to hold, it suffices that $n(x)$ be not divisible by x, however, since $n(x)$ is an irreducible polynomial over the field GF(2), this will always be case. Since $r(x)$ and $n(x)$ are relatively prime, there will be two polynomials $r^{-1}(x)$ and $n'(x)$ with the property that

$$r(x)r^{-1}(x) + n(x)n'(x)=1,$$

where $r^{-1}(x)$ is the inverse of $r(x)$ modulo $n(x)$. The polynomials $r^{-1}(x)$ and $n'(x)$ can be computed using the extended Euclidean algorithm[1]. The Montgomery multiplication of $a(x)$ and $b(x)$ is defined as the product

$$c(x)= a(x)b(x)r^{-1}(x) \bmod n(x),$$

which can be computed using the following algorithm[11] :

Algorithm 1. MMM($a(x)$, $b(x)$)
Bit-Level Algorithm for Montgomery Multiplication
Input : $a(x)$, $b(x)$, $n(x)$
Output : $c(x)= a(x)b(x)x^{-m} \bmod n(x)$
Step 1 : $c(x)=0$
Step 2 : for $i=0$ to m-1
Step 3 : $c(x)=c(x)+a_i b(x)$
Step 4 : $c(x)=c(x)+c_0 n(x)$
Step 5 : $c(x)=c(x)/x$

3.2 Modular Exponentiation Algorithm

The exponentiation algorithm is usually employed to compute $m(x)^E \bmod n(x)$, where E can be expressed as $E=[e_{m-1}, e_{m-2}, \ldots, e_1, e_0]$, and $e_i \in \{0,1\}$. First, the computation of $m(x)^E \bmod n(x)$ is divided into the LSB-first method or MSB-first method according to the method used to process the exponent E, $[e_{m-1}, e_{m-2}, \ldots, e_1, e_0]$.

LSB-first exponentiation computes E in the order of e_0 to e_{m-1}. So, the computation of M^E is as follows :

$$M^E = M^{e_0}(M^2)^{e_1}(M^4)^{e_2}\ldots(M^{2^{m-1}})^{e_{m-1}}$$

The LSB first method can be used to compute modular squaring and modular multiplication concurrently. The algorithm for LSB-first exponentiation using the Montgomery multiplication algorithm is as follows [10]:

Algorithm 2. MME($m(x)$, E, R_2)
LSB-first Exponentiation Algorithm using Montgomery multiplication
Input : $m(x)$, E, $n(x)$, $R_2 = r(x)r(x) \bmod n(x)$
Output : $c(x) = m(x)^E \bmod n(x)$
Step 1 : $\overline{m}(x) = \mathrm{MMM}(m(x), R_2)$
Step 2 : $\overline{c}(x) = \mathrm{MMM}(1, R_2)$
Step 3 : for i=0 to m-1
Step 4 : if e_i ==1 then $\overline{c}(x) = \mathrm{MMM}(\overline{m}(x), \overline{c}(x))$
Step 5 : $\overline{m}(x) = \mathrm{MMM}(\overline{m}(x), \overline{m}(x))$
Step 6 : $c(x) = \mathrm{MMM}(\overline{c}(x), 1)$

4 Cellular Automata Design

This chapter proposes a structure within which the multiplication of the structure of $\mathrm{MMM}(\overline{m}(x), \overline{c}(x))$ and $\mathrm{MMM}(\overline{m}(x), \overline{m}(x))$ and Steps 4 and 5 of Algorithm 2, which are the essential elements for obtaining $c(x) = m(x)^E \bmod n(x)$ over GF(2^m), can be performed in a short time using a cellular automata.

According to Algorithm 2, to compute LSB-first exponentiation, it is necessary to compute $\mathrm{MMM}(\overline{m}(x), \overline{c}(x))$, which is the modular multiplication used in Step 4, and $\mathrm{MMM}(\overline{m}(x), \overline{m}(x))$, which is the squaring used in Step 5. Because there is no data dependency between Steps 4 and 5 in Algorithm 2, $\mathrm{MMM}(\overline{m}(x), \overline{c}(x))$ and $\mathrm{MMM}(\overline{m}(x), \overline{m}(x))$ can be computed concurrently.

These two computations can be expressed as follows:

$$\begin{aligned}\mathrm{MMM}(\overline{m}(x), \overline{c}(x)) &= \overline{m}(x)\,\overline{c}(x)r^{-1} \bmod n(x)\\ &= \overline{m}(x)\,\overline{c}(x)x^{-m} \bmod n(x)\\ &= \overline{m}(x)(\overline{c_{m-1}}\,x^{m-1} + \overline{c_{m-2}}\,x^{m-2} + \ldots + \overline{c_1}\,x + \overline{c_0})\,x^{-m} \bmod n(x)\end{aligned}$$

$$= (\overline{m}(x)\overline{c_{m-1}}\,x^{-1} \bmod n(x) + (\overline{m}(x)\overline{c_{m-2}}\,x^{-2} \bmod n(x) + \dots + \overline{m}(x)\overline{c_1}\,x^{-m+1} \bmod n(x) + \overline{m}(x)\overline{c_0}\,x^{-m} \bmod n(x)) \bmod n(x) \quad (1)$$

$$\begin{aligned}\mathrm{MMM}(\overline{m}(x), \overline{m}(x)) &= \overline{m}(x)\overline{m}(x)r^{-1} \bmod n(x) \\ &= \overline{m}(x)\overline{m}(x)x^{-m} \bmod n(x) \\ &= \overline{m}(x)(\overline{m_{m-1}}\,x^{m-1} + \overline{m_{m-2}}\,x^{m-2} + \dots + \overline{m_1}\,x + \overline{m_0})\,x^{-m} \bmod n(x) \\ &= (\overline{m}(x)\overline{m_{m-1}}\,x^{-1} \bmod n(x) + (\overline{m}(x)\overline{m_{m-2}}\,x^{-2} \bmod n(x) + \dots \\ &\quad + \overline{m}(x)\overline{m_1}\,x^{-m+1} \bmod n(x) + \overline{m}(x)\overline{m_0}\,x^{-m} \bmod n(x)) \bmod n(x) \quad (2)\end{aligned}$$

In equations (1) and (2), $\overline{m}(x)x^{-1} \bmod n(x)$, $\overline{m}(x)x^{-2} \bmod n(x)$, …, $\overline{m}(x)x^{-m+1} \bmod n(x)$, and $\overline{m}(x)x^{-m} \bmod n(x)$ are the common parts in the modular squaring and multiplication. The common-multiplicand multiplication algorithm used to calculate $\mathrm{MMM}(\overline{m}(x), \overline{c}(x))$ and $\mathrm{MMM}(\overline{m}(x), \overline{m}(x))$ at the same time is represented as follows :

Algorithm 3 : MMMS($\overline{m}(x)$, $\overline{c}(x)$, $n(x)$)

Montgomery Algorithm for Computing Multiplication and Squaring

Input : $\overline{m}(x)$, $\overline{c}(x)$, $n(x)$

Output : $M(x)=\mathrm{MMM}(\overline{m}(x), \overline{c}(x))$, $S(x)=\mathrm{MMM}(\overline{m}(x), \overline{m}(x))$

Step1 : $t(x)=\overline{m}(x)$

Step2 : $M(x)=0$ $S(x)=0$

Step3 : **for** $i=m-1$ **to** 0

Step4 : $t(x)=(t(x)+t_0 n(x))/x$

Step5 : $M(x)= M(x)+t(x)\overline{c_i}$, $S(x)= S(x)+ t(x)\overline{m_i}$

The value $t(x)$, which is the common part of the multiplication and squaring, is computed once in Step 4, and then used to compute $M(x)$ and $S(x)$ in Step 5 at the same time.

Therefore, a structure is proposed in which modular multiplication and squaring can be computed simultaneously over GF(2^m) based on a cellular automata through computing $t(x)$, which is the common part in equations (1) and (2) and step 4 of Algorithm 3, only once, using the result, and then obtaining the remaining part of Step 5 in parallel.

The basic operations for implementing Algorithm 3 are as follows:

For the common part of the multiplication and squaring:

Operation 1 : $t(x)=(t(x)+t_0n(x))/x$

For the remaining part of the multiplication :

Operation 2 : $M(x)= M(x)+t(x)\overline{c_i}$

For the remaining part of the squaring :

Operation 3 : $S(x) = S(x) + t(x)\overline{m_i}$

In order to perform Operation 1($t(x)=(t(x)+t_0n(x))/x$), first, the operation $t(x)=t(x)+t_0n(x)$ must be performed if the LSB of $t(x)$ is 1, i.e., if $t_0=1$, the XOR computation of (t_{m-1},…, t_2, t_1, t_0) and (n_{m-1},…, n_2, n_1, n_0) is performed according to each bit.

Its structure is shown in Fig. 1 as follows:

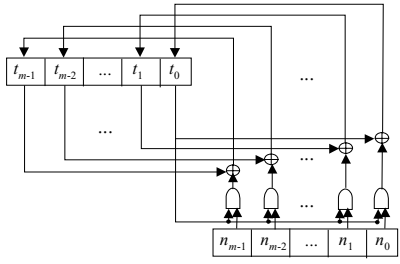


Fig. 1. Structure of $t(x)=t(x)+t_0n(x)$.

Next, the operation $t(x)=t(x)/x$ is performed. For this operation a 1-dimensional PBCA with m cells is used. The value of the left neighbor for each cell is received as its own value for the characteristic matrix, and the $m \times m$ matrix B, as shown below, is used since the leftmost cell and rightmost cell have the characteristics of a neighboring PBCA for a cyclic right shift

$$B = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 & 1 \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & 0 \end{bmatrix}$$

A cellular automata with such a characteristic matrix is under rule 240, for which the structure is as follows:

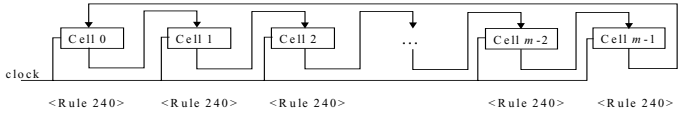


Fig. 2. PBCA structure with characteristic matrix B

Hereinafter, it is supposed that a CA with the structure in Fig. 2 is a PBCA with a characteristic matrix B.

In Fig.3, the $t(x)$ register can be replaced by a CA. Yet in Operation 1, a one bit right shift operation must occur after the modular reduction. This process is as follows.

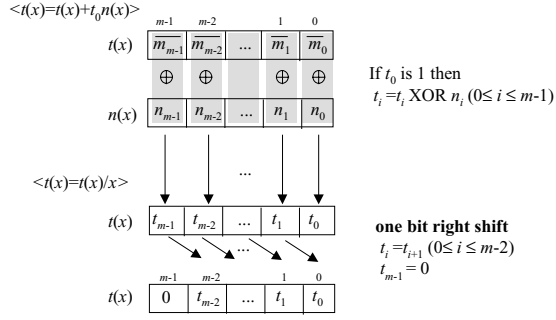


Fig. 3. Process of one bit right shift after modular reduction

In the current study, to accomplish a one bit right shift operation using a CA, $\overline{m(x)}$ of the initial value of $t(m)$ is inputted as the CA's initial value. Then the one bit cyclic right shift process is operated using the PBCA. Thereafter, the output of the CA becomes ($\overline{m_0}$, $\overline{m_{m-1}}$, $\overline{m_{m-2}}$,…, $\overline{m_1}$). At this time, if the value of MSB is 1, $\overline{m_0}$ =1, a modular reduction will take place. As such, the value of MSB is used to determine whether or not a modular reduction will occur, and the real value is filled with 0, as in Fig 3. Using these methods the process for accomplishing Operation 1 is the same as in Fig 4.

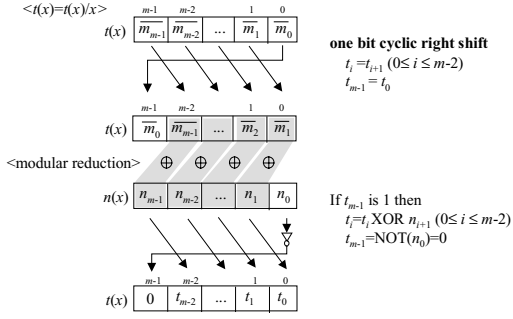


Fig. 4. Process of modular reduction after one bit cyclic right shift

The structure of Operation 1 using Fig 4 is shown in Fig 5.

In order to perform Operation 2, $M(x) = M(x) + t(x)\overline{c_i}$ for m-1≥i≥0, the $t(x)\overline{c_i}$ operation is first reviewed. For a $t(x)$ $\overline{c_i}$ operation with m-1≥i≥0, the m bits of the $t(x)$ register are inputted into one of the m AND gates and into the remaining $\overline{c_i}$. The next result and $M(x)$ are subject to an XOR, and the result is then stored again at $M(x)$. The

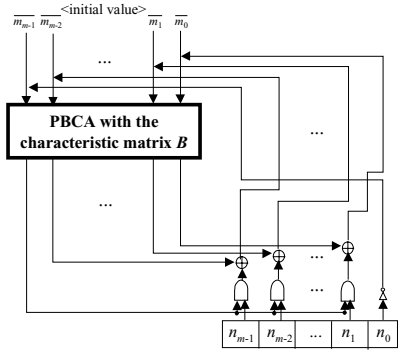


Fig. 5. Structure of Operation 1($t(x)=(t(x)+t_0n(x))/x$).

structure for this is shown in Fig. 6, which presents the computation in the *ith* clock for m-1≥i≥0.

To perform Operation 3, for squaring, $\overline{c}(x)$ is substituted with $\overline{m}(x)$ in the modular multiplication.

Now, the structure shown in Fig. 7, in which modular multiplication and squaring using the Montgomery algorithm can be performed simultaneously using a CA, is created by combining the construction diagrams proposed in Figs. 5 and 6.

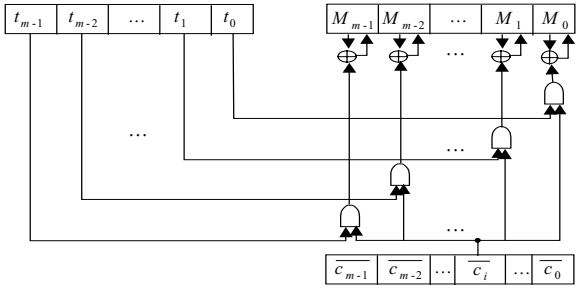


Fig. 6. Structure of operation 2 ($M(x) = M(x) + t(x)\ \overline{c_i}$ for m-1≥i≥0).

Fig. 7 shows the computation at the *ith* clock for m-1≥i≥0, where the initial value is set using a PBCA with the characteristic matrix B and $\overline{c}$, $\overline{m}$, M, S, N register prior to the first operation. Plus, the $t(x)$ register in Fig. 6 can be replaced by the CA. Each initial value is as follows:

- Initial values of PBCA : $\overline{c(x)} = \overline{c_{m-1}}\ \overline{c_{m-2}} \cdots \overline{c_1}\ \overline{c_0}$
- Initial values of $\overline{c}$ register : $\overline{c(x)} = \overline{c_{m-1}}\ \overline{c_{m-2}} \cdots \overline{c_1}\ \overline{c_0}$
- Initial values of $\overline{m}$ register : $\overline{m(x)} = \overline{m_{m-1}}\ \overline{m_{m-2}} \cdots \overline{m_1}\ \overline{m_0}$
- Initial values of M register : all 0

- Initial values of S register : all 0
- Initial values of N register : $n(x) = n_{m-1} \ldots n_2 n_1 n_0$

As a result, it is possible to simultaneously perform multiplication and squaring using the Montgomery algorithm in m clock cycles using m cells, $3m$-1 AND gates, $3m$-1 XOR gates, 1 NOT gate, and 5 registers if the structure shown in Fig. 7 is used

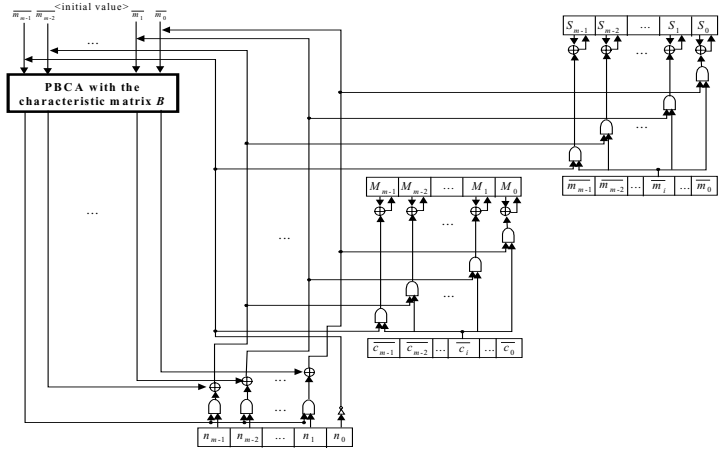


Fig. 7. Structure of simultaneous performance of modular multiplication and squaring using CA.

Table 1. Comparison of multiplication and squaring using Montgomery algorithm

Structure	Systolic array (1-dimensional)	Cellular Automata
	LEE *et al* [10]	Proposed paper
Operation	Montgomery multiplication and squaring over GF (2^m)	Montgomery multiplication and squaring over GF (2^m)
NO. of cells	m+1	m
NO. of AND gates	3m+3	3m-1
NO. of XOR gates	3m+3	3m-1
NO. of NOTgates	0	1
NO. of latches	14m+14	0
NO.of MUXes	3m+3	0
NO. of control signals	1	0
NO.of registers	0	5
Execution time (clock cycles)	3m	m

5 Conclusion

The current paper proposed a new structure in which modular multiplication and squaring using the Montgomery algorithm can be processed simultaneously for effective exponentiation over GF(2^m) based on a 3-neighbor cellular automata. The proposed structure makes use of the common-multiplicand concept in the Montgomery algorithm for computing the multiplication and squaring simultaneously. As a result, it is possible to perform multiplication and squaring in m clock cycles using m cells, $3m$-1 AND gates, $3m$-1 XOR gates, 1 NOT gate, and 5 registers.

The performance of the proposed architecture was compared with that of a previous study, as shown in Table 1.

In conclusion, the structure proposed in the current paper is much more efficient in terms of space and time than [10]. In addition, modular exponentiation, division, inversion architecture, etc. can also be efficiently implemented based on the Montgomery multiplier/squarer proposed in this paper.

References

1. R.J. McEliece, *Finite Fields for Computer Scientists and Engineers*, New York:Kluwer Academic, 1987
2. W. Diffie and M.E. Hellman, "New directions in cryptography," *IEEE Trans. on Info. Theory*, VOL. 22, pp.644-654, Nov. 1976.
3. T. ElGamal. "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Trans. on Info. Theory*, Vol. 31(4). pp. 469-472, July 1985.
4. A.J. Menezes, *Elliptic Curve Public Key Cryptosystems*, Kluwer Academic Publishers,1993.
5. C.–S. YEH, IRVING S. REED, T.K. TRUONG, "Systolic Multipliers for Finite Fields GF(2^m)," *IEEE Transactions On Computers*, Vol. C-33, No. 4, pp. 357-360, April 1984.
6. C.L. Wang, J.L. Lin, "Systolic Array Implementation of Multipliers for Finite Fields GF(2^m)," *IEEE Transactions On Circuits And Systems*, VOL. 38, NO. 7, pp. 796-800,July 1991.
7. P.L. Montgomery, "Modular multiplication without trial division," *Mathematics of Computation*, 44(170):519-521, April, 1985.
8. M. Delorme, J. Mazoyer, *Cellular Automata*, Kluwer academic Publishers 1999.
9. Stephen Wolfram, *Cellular Automata and Complexity*, Addison-Wesly Publishing Company, 1994.
10. W.H. Lee, K.J. Lee, K.Y. Yoo, "Design of a Linear Systolic Array for Computing Modular Multiplication and Squaring in GF(2^m)," *Computer and Mathematics with Applications* 42, pp. 231-240, 2001.
11. Ç. K. Koç, T. Acar, "Montgomery Multiplication in GF(2^k)," Kluwer Academic Publishers, Designs, Codes and Cryptography, 14(1), pp. 57-69, April 1998.
12. Knuth, *The Art of Computer programming*, Vol. 2/Seminumerical Algorithms, Addison-Wesley, 1969.

A Decision Tree Algorithm for Distributed Data Mining: Towards Network Intrusion Detection

Sung Baik [1] and Jerzy Bala [2]

[1] Sejong University, Seoul 143-747, KOREA
sbaik@sejong.ac.kr
[2] Datamat Systems Research, Inc.
1600 International Drive, McLean, VA 22102, USA
jbala@dsri.com

Abstract. This paper presents preliminary works on an agent-based approach for distributed learning of decision trees. The distributed decision tree approach is applied to intrusion detection domain, the interest of which is recently increasing. In the approach, a network profile is built by applying a distributed data analysis method for the collection of data from distributed hosts. The method integrates inductive generalization and agent-based computing, so that classification rules are learned via tree induction from distributed data to be used as intrusion profiles. Agents, in a collaborative fashion, generate partial trees and communicate the temporary results among them in the form of indices to the data records. Experimental results are presented for military network domain data used for the network intrusion detection in KDD cup 1999. Several experimental results show that the performance of distributed version of decision tree is much better than that of non-distributed version with data collected manually from distributed hosts.

1 Introduction

Recently, the amount of data has rapidly been increased and the related data are naturally located at geographically distributed sites. However, most of the existing knowledge discovery/data mining tools assume that all the data can be found in a single homogeneous database, and the vast amounts of data are collected into the centralized sites in a manual way for data analysis in today's real business world.

On the other hands, in the research community, there have been some efforts directed towards Distributed Data Mining (DDM), which has been applied to a variety of industry fields. The examples of its application are as follows:

1. It is recently very important and urgent to share medical information in distributed medical/healthcare data repositories located in local hospitals and medical centers to forecast of the possible geographical impact of the epidemic. Through the shared medical knowledge discovered by a distributed data mining [1], healthcare professionals and managers can be provided with medical strategic-planning services such as *analyzing treatment pattern, analyzing outcomes of treatment, analyzing cost-effectiveness of health care, forecasting*

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 206–212, 2004.
© Springer-Verlag Berlin Heidelberg 2004

2. *'new disease' and strategizing appropriate preventive measures,* and *forecasting the spread of infectious diseases* [2].

3. A variety of data and information existed at distributed sites in an e-commerce environment are shared and analyzed to provide customers with an on-line shopping center on the web. A distributed data mining system [3] for e-commerce environments provides business sites such as individual vendors and on-line traders with ASP based data mining system, rather than they have their own data mining systems.

4. As credit card transactions are increasing rapidly, it is almost impossible that the massive amounts of transaction data are moved into a centralized repository for on-line data analysis. Therefore, it is very vital that a large number of data in each distributed site is analyzed separately and each analyzed fraud model is shared with each other for prompt credit card fraud detection [4].

The additional examples of its applications to exploit the full benefit of distributed computation with communication are introduced in [5]. This paper presents an agent based distributed data mining approach, for tree induction from vertically partitioned data sets and its application to the computer network intrusion detection area. Intrusion detection is an important component of a modern computer network system's protection from unauthorized and hostile use. The main purpose of a typical intrusion detection system is to detect outside intruders as well as inside intruders that may break into the system (based on observations of deviations from normal system usage patterns). This is achieved through building a statistical pattern based profile of a monitored system. In typical intrusion detection scenarios, this profile is built from data collected from multiple hosts.

2 Distributed Data Mining Approach

In the past, some research works had been achieved on traditional data mining algorithms such as association rules [6], Bayesian learning approach [7] and clustering in a distributed database environment. Recently, the implementation environments, for the distributed data mining, such as grid environments [8-11] and agent-based architectures [12,13], have been developed. However, even though a variety of data in real world are located in the heterogeneous data sites[1], most of researches for the distributed data mining are for analyzing data from homogeneous data sites where the distributed databases are a horizontally partitioned and the database schema in every site are same.

In this paper, we present a decision tree algorithm in a distributed database environment and a distributed data mining system, which has an inter-agent communication mechanism on the agent based framework implemented in Java development environments. The decision tree algorithm for the distributed data

[1] In the heterogeneous data sites, the distributed databases are a vertically partitioned and the database schemas in every site are different.

mining has been developed from a traditional decision tree algorithm [14], for a centralized data repository, which is one of the most popular data mining algorithms.

In order to fully take advantage of all the available data, this distributed data mining system has a mechanism for integrating data from a wide variety of data sources and is able to handle data characterized by geographic (or logical) distribution, complexity and multi feature representations, and vertical partitioning/distribution of feature set.

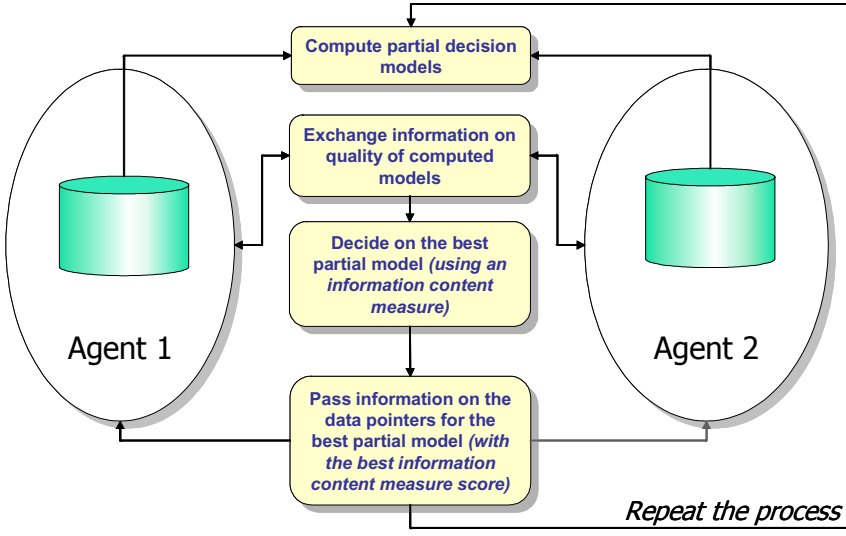


Fig. 1. The basic concept behind the distributed data mining approach

The basic concept behind the distributed data mining approach presented in this paper is depicted in Figure 1. Distributed mining is accomplished via a synchronized collaboration of Agents and a Mediator component facilitates the communication among Agents. Distributed data mining results in a set of rules generated through a tree induction algorithm. The tree induction algorithm, in an iterative fashion, determines the feature which is most discriminatory and then it dichotomizes (splits) the data into classes categorized by this feature [14]. The next significant feature of each of the subsets is then used to further partition them and the process is repeated recursively until each of the subsets contains only one kind of labeled data. The resulting structure is called a decision tree, where nodes stand for feature discrimination tests, while their exit branches stand for those subclasses of labeled examples satisfying the test. A tree is rewritten to a collection of rules, one for each leaf in the tree. Every path from the root of a tree to a leaf gives one initial rule. The left-hand side of the rule contains all the conditions established by the path, and the right-hand side specifies the classes at the leaf. Each such rule is simplified by

removing conditions that do not seem helpful for discriminating the nominated class from other classes

3 The Communication between Agents

The Distributed Data Mining (DDM) component includes a number of Data Mining Agents whose efforts are coordinated through a facilitator. One of the major functions of the facilitator is to collect information from various DM Agents and to broadcast the collected information to other Agents involved in the mining process. To this end, there is a certain amount of cost associated with the distributed mining process, namely that of the communication bandwidth. For very large datasets, the high cost of transferring information from one Agent to another can become a major bottleneck in the data mining process.

Our goal has been to reduce the inter-Agent communication bandwidth by finding ways to reduce the amount of information necessary for agent collaboration. As explained previously, each DM Agent in our architecture is responsible for mining its own local data by finding the feature (or attribute) that can best split the data records into the various training classes (i.e. the attribute with the highest information gain). The selected attribute is then sent as a candidate attribute to the Mediator for overall evaluation.

Once the Mediator has collected the candidate attributes of all the Agents, it can then select the attribute with the highest information gain as the winner. The winner Agent (i.e. the Agent whose database includes the attribute with the highest information gain) will then continue the mining process by splitting the data records using the winning attribute and its associated split value. This split results in the formation of two separate clusters of data records (i.e. those satisfying the split criteria and those not satisfying it). The associated indices of the data records in each cluster are passed to the Mediator to be used by all the other agents. The other (i.e. non-winning) Agents access the index information passed to the Mediator by the winning Agent and split their data records accordingly. The mining process then continues by repeating the process of candidate feature selection by each of the Agents.

Thus, the bulk of the information which needs to be passed from one DM Agent to another during the collaborative mining process is comprised of a list of data record indexes. Passing the index information using an integral representation can become a major problem for a large number of data records. To this end, we have employed an index bit-vector generation technique.

During the index bit-vector generation phase, the index information, normally represented as a set of integers (i.e. record numbers), is converted to a bit-vector representation. In a bit-vector representation, each individual bit corresponds to the index of a single data record. Thus bit number three, for example, corresponds to the index of the third data record. The actual value of the bit represents the presence or absence of the corresponding data record in the data cluster being passed from one DM agent to another. A value of "1" represents the presence of that data record, while a value of "0" represents its absence. This representation is much more compact than the set-of-integers representation. The difference in size for large number of data

records is dramatic. To further reduce the size of the data being transferred, a bit-vector representation is compressed.

4 Experimental Validation

4.1 Experimental Data

The data set used in experiments originates from the Third International Knowledge Discovery and Data Mining Tools Competition [15], the task of which was to build a network intrusion detector, a predictive model capable of distinguishing between two classes. One class is “bad” connections, called intrusions or attacks, and the other one is “good” normal connections. The data set contains a standard set of data to be audited, which includes a wide variety of intrusions simulated in a military network environment. The raw training data was about four gigabytes of a compressed binary TCP data dump from seven weeks of network traffic. This was processed into about five million connection records. Similarly, the two weeks of test data yielded around two million connection records. A connection is a sequence of TCP packets starting and ending at well defined times, between which data flows to and from a source IP address to a target IP address under some well defined protocol. Each connection is labeled as either normal, or as an attack, with exactly one specific attack type. Each connection record consists of about 100 bytes. There are several categories of derived features. The “same host” features examine only the connections in the past two seconds that have the same destination host as the current connection, and calculates statistics related to protocol behavior, service, etc. The similar ``same service" features examine only the connections in the past two seconds that have the same service as the current connection. "Same host" and "same service" features are together called time-based traffic features of the connection records.

4.2 Experiments

Experiments were conducted to generate data pattern for two classes (i.e., an attack and a normal state categories). The data used in experiments represent a set of data chunks collected from distributed host machines. Various comparisons have revealed exactly the same rule sets for both distributed and non-distributed versions of the data-mining algorithm. To estimate the processing time for rule extraction, we have built databases with same number (100) of fields and the increasing number of records. To clearly compare the performance of distributed version with that of the non-distributed version, only two agents with two processes (with 1.7 GHZ CPU and 516MB RAM) have been used. Figure 2 presents the summary of processing time for decision rule extraction. In experimentation steps, the indices from 1 to 8 represent $5x10^4$, $1x10^5$, $1.5x10^5$, $2x10^5$, $2.5x10^5$, $3x10^5$, $3.5x10^5$, and $4x10^5$ records, respectively. According to several experimental results, processing time monotonically increases with changing number and the performance of non-distributed version is lower than that of distributed version, even though there are some overhead in the distributed version.

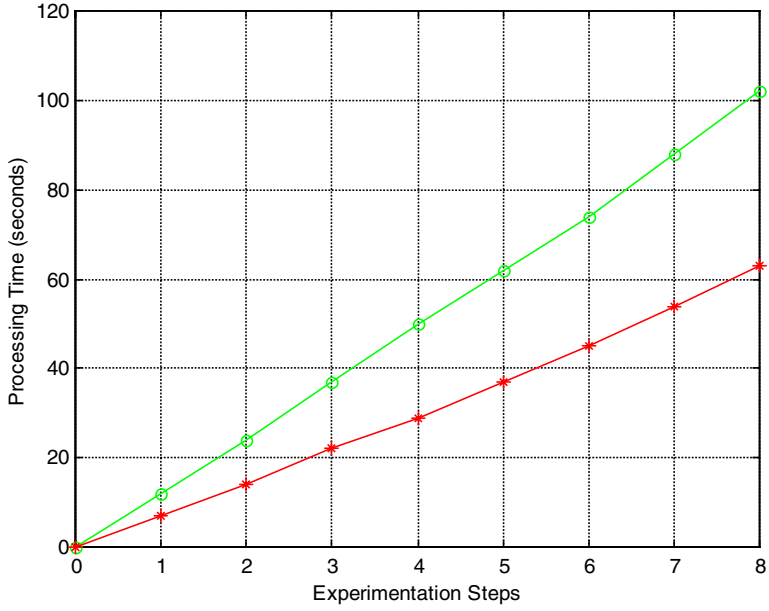


Fig. 2. The processing time for decision rule extraction ('o' and '*' marks indicate non-distributed version and distributed version, respectively.)

5 Conclusion and Future Work

This paper provided a significant approach for distributed learning of decision trees in the intrusion detection domain. The approach is to integrate inductive generalization and agent-based computing so that classification rules can be learned via tree induction from distributed data to be used as intrusion profiles.

Since the current system is implemented to support only two agents, we presented experimental results with data from two distributed hosts and compared them with those of the non-distributed version. As a future work, we need the implementation of additional agents so that we can experiment the network intrusion detection data from several distributed hosts.

References

1. Zaidi, S.Z.H., Abidi, S.S.R. and Manickam, S., Distributed data mining from heterogeneous healthcare data repositories: towards an intelligent agent-based framework, In: Proceedings of the 15th IEEE Symposium on Computer-Based Medical Systems (CBMS), pp.339 -342, 2002

2. Abidi, S.S.R., Applying Knowledge Discovery in Healthcare: An Info-Structure for Delivering Knowledge-Driven Strategic Services, Medical Informatics Europe'99, IOS Press, pp. 453-456, 1999.
3. Krishnaswamy, S., Zaslavsky, A. and Loke, S.W. , An architecture to support distributed data mining services in e-commerce environments, In: Proceedings of Second International Workshop on Advanced Issues of E-Commerce and Web-Based Information Systems (WECWIS) , pp.239 -246, 2000
4. Chan, P.K., Fan, W., Prodromidis, A.L. and Stolfo, S.J., Distributed data mining in credit card fraud detection, Intelligent Systems, IEEE [see also IEEE Expert] , Vol.14 Issue:6, pp.67 -74, 1999.
5. Kargupta, H. and Park, B., Collective Data Mining: A New Perspective toward Distributed Data Analysis, In H. Kargupta and P. Chan, editors, Advanced in Distributed and Parallel Knowledge Discovery, AAAI/MIT Press, pp 133-184., 2000.
6. Cheung, D.W., Ng, V.T., Fu, A.W., Yongjian Fu, Efficient mining of association rules in distributed databases , In: Proceedings of the IEEE Transactions on Knowledge and Data Engineering , Vol 8, Issue: 6 , pp. 911 -922, 1996
7. Yamanish, K., Distributed cooperative Bayesian learning strategies, In: Proceedings of COLT 97 (ACM), pp.250-262, 1997.
8. Grossman, R.L., Yunhong Gu, Hanley, D., Xinwei Hong, Levera, J., Mazzucco, M., Lillethun, D., Mambretti, J. and Weinberger, J., Mass Storage Systems and Technologies (MSST), In: Proceedings of the 20th IEEE/11th NASA Goddard Conference, 2003, 62 -66
9. Wei Du; Agrawal, G., Developing distributed data mining implementations for a grid Environment , Cluster Computing and the Grid 2nd IEEE/ACM International Symposium (CCGRID), pp. 410 -411, 2002
10. Mario Cannataro, Domenico Talia and Paolo Trunfio, Distributed data mining on the grid, Future Generation Computer Systems, Vol. 18, Issue : 8, pp.1101-1112, 2002
11. Cannataro, M., Clusters and Grids for Distributed and Parallel Knowledge Discovery, Lecture Notes in Computer Science, Vol. 1823, pp.708 – 715, 2000
12. Zaidi, S.Z.H., Abidi, S.S.R. and Manickam, S., Distributed data mining from heterogeneous healthcare data repositories: towards an intelligent agent-based framework, In: Proceedings of the 15th IEEE Symposium on Computer-Based Medical Systems (CBMS), pp.339 -342, 2002
13. Klusch, M., Lodi, S. and Moro, G., Agent-Based Distributed Data Mining: The KDEC Scheme, Lecture Notes in Artificial Intelligent, Vol. 2586, pp.104 – 122, 2003
14. Quinlan, J. R., and Rivest, R.L., Inferring Decision Trees Using the Minimum Description Length Principle, Information and Computation, Vol. 80, no. 3, 1989.
15. See Web site at http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

Maximizing Parallelism for Nested Loops with Non-uniform Dependences

Sam Jin Jeong

Division of Information and Communication Engineering, Cheonan University
Anseo-dong 115, Cheonan City, Korea 330-704
sjjeong@cheonan.ac.kr

Abstract. Partitioning of loops is a very important optimization issue and requires the efficient and exact data dependence analysis. Although several methods exist in order to parallelize loops with non-uniform dependences, most of them perform poorly due to irregular and complex dependence constraints. This paper proposes Improved Region Partitioning Method for minimizing the size of the sequential region and maximizing parallelism. Our approach is based on the Convex Hull theory that has adequate information to handle non-uniform dependences. By parallelizing anti dependence region using variable renaming, we will divide the iteration space into two parallel regions and one or less sequential region. Comparison with other schemes shows more parallelism than the existing techniques.

1 Introduction

Given a sequential program, a challenging problem for parallelizing compilers is to detect maximum parallelism. It is generally agreed upon, and shown in the study by Kuck and et al. [1] that most of the computation time is spend in loops. Current parallelizing compilers pay much of their attention to loop parallelization [2]. A lot of work has been proposed for parallelizing uniform dependence loops, such as loop interchange, loop permutation, skew, reversal, wavefront, tiling, etc [3],[4],[5].

According to an empirical study [7], nearly 66% of the array references have linear or partially linear subscript expressions and 45% of two-dimensional array references are coupled and most of these lead to non-uniform dependences.

Some parallelization techniques, based on Convex Hull theory [8] which has been proved to have enough information to handle non-uniform dependences, are minimum dependence distance tiling [9], the unique set oriented partitioning [10] and three region partitions [11]. This paper will focus on parallelization of flow and anti dependence loops with non-uniform dependences.

The rest of this paper is organized as follows. Section two describes our loop model and introduces the concept of Complete Dependence Convex Hull(CDCH). Section three presents our Improved Region Partitioning technique. Section four compares our proposed technique with related techniques. Finally, we conclude in section five.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 213–222, 2004.
© Springer-Verlag Berlin Heidelberg 2004

Example l.

```
do i = 1, 10
  do j = 1, 10
    A(2*j+3, i+j+5) = . . .
          . . . = A(2*i+j-1, 3*i-1)
  enddo
enddo
```

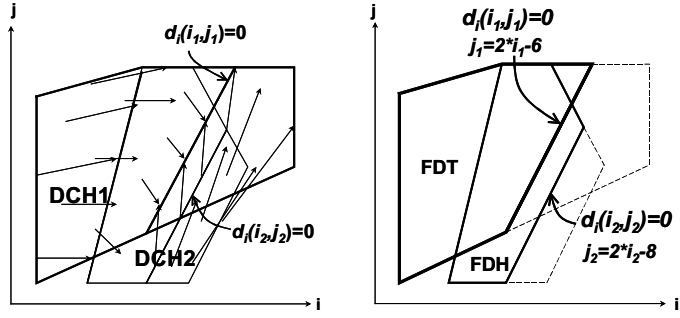


Fig. 1. (a) CDCH (b) FDT and FDH of Example 1

2 Program Model and Dependence Analysis

The loop model considered in this paper is doubly nested loops with linearly coupled subscripts and both lower and upper bounds for loop variables should be known at compile time. The loop model has the form in Fig. 2, where $f_1(I, J)$, $f_2(I, J)$, $f_3(I, J)$, and $f_4(I, J)$ are linear functions of loop variables.

```
do I = l1, u1
  do J = l2, u2
       A(f1(I, J), f2(I, J)) = . . .
      . . . = A(f3(I, J), f4(I, J))
  enddo
enddo
```

Fig. 2. A doubly nested loop model

The loop in Fig. 2 carries cross iteration dependences if and only if there exist four integers (i_1, j_1, i_2, j_2) satisfying the system of linear diophantine equations given by (1) and the system of inequalities given by (2). The general solution to these equations can be computed by the extended GCD [6] or the power test algorithm [12] and forms a **DCH** (Dependence Convex Hull).

$$f_1(i_1, j_1) = f_3(i_2, j_2) \text{ and } f_2(i_1, j_1) = f_4(i_2, j_2) \tag{1}$$

$$l_1 \quad i_1, i_2 \quad u_1 \text{ and } l_2 \quad j_1, j_2 \quad u_2 \tag{2}$$

From (1), (i_1, j_1, i_2, j_2) can be represented as

$(i_1, j_1, i_2, j_2) = (g_1(i_2, j_2), g_2(i_2, j_2), g_3(i_1, j_1), g_4(i_1, j_1))$
where g_i are linear functions.

From (2), two sets of inequalities can be written as

$$L_1 \le i_1 \le u_1 \text{ and } l_2 \le j_1 \le u_2 \text{ and} \tag{3}$$
$$l_1 \quad g_3(i_1, j_1) \quad u_1 \text{ and } l_2 \quad g_4(i_1, j_1) \quad u_2$$

$$L_1 \le i_2 \le u_1 \text{ and } l_2 \le j_2 \le u_2 \text{ and} \tag{4}$$
$$l_1 \quad g_1(i_2, j_2) \quad u_1 \text{ and } l_2 \quad g_2(i_2, j_2) \quad u_2$$

And, (3) and (4) form DCHs denoted by DCH1 and DCH2, respectively [10]. Clearly, if we have a solution (i_1, j_1) in DCH1, we must have a solution (i_2, j_2) in DCH2, because they are derived from the same set of equations (1). The union of DCH1 and DCH2 is called Complete DCH (**CDCH**), and all dependences lie within the CDCH. Fig. 1(a) shows the CDCH of Example 1, which is given in [13].

If iteration (i_2, j_2) is dependent on iteration (i_1, j_1), then we have a dependence vector $d(i_1, j_1) = (d_i(i_1, j_1), d_j(i_1, j_1)) = (i_2\text{-}i_1, j_2\text{-}j_1)$

So, for DCH1, we have

$$d_i(i_1, j_1) = g_3(i_1, j_1) - i_1 = (\alpha_{11} - 1)i_1 + \beta_{11}j_1 + \gamma_{11} \text{ and} \tag{5}$$
$$d_j(i_1, j_1) = g_4(i_1, j_1) - j_1 = \alpha_{12}i_1 + (\beta_{12} - 1)j_1 + \gamma_{12}$$

For DCH2, we have

$$d_i(i_2, j_2) = i_2 - g_1(i_2, j_2) = (1 - \alpha_{21})i_2 - \beta_{21}j_2 - \gamma_{21} \text{ and} \tag{6}$$
$$d_j(i_2, j_2) = j_2 - g_2(i_2, j_2) = -\alpha_{22}i_2 + (1 - \beta_{22})j_2 - \gamma_{22}$$

We can write these dependence distance functions in a general form as

$$d(i_1, j_1) = (d_i(i_1, j_1), d_j(i_1, j_1)) \tag{7}$$
$$d(i_2, j_2) = (d_i(i_2, j_2), d_j(i_2, j_2))$$
$$d_i(i_1, j_1) = p_1 * i_1 + q_1 * j_1 + r_1$$
$$d_j(i_1, j_1) = p_2 * i_1 + q_2 * j_1 + r_2$$
$$d_i(i_2, j_2) = p_3 * i_2 + q_3 * j_2 + r_3$$
$$d_j(i_2, j_2) = p_4 * i_2 + q_4 * j_2 + r_4$$

where p_i, q_i, and r_i are real values and i_1, j_1, i_2, and j_2 are integer variables of the iteration space.

The properties of DCH1 and DCH2 can be found in [10].

3 Improved Region Partitioning Method

In this section, we present an improved method to partition doubly nested loops with flow and anti dependence sets. By variable renaming, we eliminate anti dependence sets from the loop. There remain only flow dependence sets in the loop. We will show how to improve parallelism in this case.

We define the flow dependence tail set (FDT) and the flow dependence head set (FDH) as follows.

Definition 1. *Let L be a doubly nested loop with the form in Fig. 2. If line $d_i(i_1, j_1) = 0$ intersects DCH1, the flow dependence tail set of the DCH1, namely **FDT**(L), is the region H, where H is equal to*

$$\text{DCH1} \cap \{(i_1, j_1) \mid d_i(i_1, j_1) \quad 0 \text{ or } d_i(i_1, j_1) \quad 0\} \tag{8}$$

Definition 2. *Let L be a doubly nested loop with the form in Fig. 2. If line $d_i(i_2, j_2) = 0$ intersects DCH2, the flow dependence head set of the DCH2, namely **FDH**(L), is the region H, where H is equal to*

$$\text{DCH2} \cap \{(i_2, j_2) \mid d_i(i_2, j_2) \quad 0 \text{ or } d_i(i_2, j_2) \quad 0\} \tag{9}$$

We also define the line LMLT (Left Most Line) and the line RMLT (Right Most Line) in FDT, and the line LMLH (Left Most Line) and the line RMLH (Right Most Line) in FDH as follows.

Definition 3. *The line that can be formed by the two left most extreme points in FDT is called the **LMLT** ($dl_i(i_1, j_1) = 0$). And the line by the two left most extreme points in FDT is called **RMLT** ($dr_i(i_1, j_1) = 0$).*

Definition 4. *The line that can be formed by the two left most extreme points in FDH is called the **LMLH** ($dl_i(i_2, j_2) = 0$). And the line by the two left most extreme points in FDH is called **RMLH** ($dr_i(i_2, j_2) = 0$).*

Property 1. *Suppose line $d_i(i, j) = p*i+q*j+r$ passes through CDCH. If $q > 0$, FDT(FDH) is on the side of $d_i(i_1, j_1)$ 0 ($d_i(i_2, j_2)$ 0), otherwise, FDT(FDH) is on the side of $d_i(i_1, j_1)$ 0 ($d_i(i_2, j_2)$ 0).*

Fig. 1(b) shows the flow dependence tail set (FDT) of DCH1 and the flow dependence head set (FDH) of DCH2 in Example 1.

In our proposed method, the Improved Region Partitioning Method, we select one or two appropriate lines among four lines such as the line LMLT and the line RMLT in FDT, and the line LMLH and the line RMLH in FDH. The line $d_i(i_1, j_1) = 0$ and the line $d_i(i_2, j_2) = 0$ are included in these four lines. The line $d_i(i_1, j_1) = 0$ is one of two lines LMLT and RMLT in FDT, and the line $d_i(i_2, j_2) = 0$ is one of two lines LMLH and RMLH in FDH. One or two selected lines divide the iteration space into two parallel regions and/or less than one serial region.

To partition the iteration space, we use the Algorithm Region_Partition, which is the algorithm of selecting the bounds in the transformed loop in two-dimensional solution space as shown in Fig. 3. The main functionality of this algorithm is to select one or two appropriate lines among four lines by position of two given lines $d_i(i_1, j_1) = 0$ and $d_i(i_2, j_2) = 0$, and two real values q_1 and q_3 given in (7). From property 1, we know that the real value q_1 determines whether the position of FDT is on side of $d_i(i_1, j_1) \geq 0$ or not, and real value q_3 determines the position of FDH. These two (or one)

selected lines are the bounds of three (or two) loops. Execution of iterations based on three (or two) regions can be obtained by transforming the original loop into three (or two) loops. If one line is selected, the iteration space is divided into two parallel regions by the selected line.

```
Algorithm Region_Partition
INPUT: four lines (LMLT, RMLT, LMLH, RMLH)
OUTPUT: two parallel regions and/or less than one serial region
BEGIN
IF (line d_i(i_1, j_1) = 0 is on the left side of line d_i(i_2, j_2) =0)
  Switch (q_1, q_3) BEGIN
     CASE 1: q_1 > 0 and q_3 > 0
        Select dl_i(i_2, j_2) = 0 (= LMLH) and d_i(i_1, j_1) = 0 (= RMLT);
        Call Transformation11(dl_i(i_2, j_2), d_i(i_1, j_1));
     CASE 2: q_1 > 0 and q_3 < 0
        /* FDH does not overlap FDT */
        Call Transformation12(d_i(i_1, j_1));
     CASE 3: q_1 < 0 and q_3 > 0
       Select d_i(i_1, j_1) = 0 (= LMLT) and d_i(i_2, j_2) = 0 (= RMLH);
       Call Transformation13(d_i(i_1, j_1), d_i(i_2, j_2));
     CASE 4: q_1 < 0 and q_3 < 0
        Select d_i(i_2, j_2) = 0 (=  LMLH) and dr_i(i_1, j_1) = 0 (= RMLT);
        Call Transformation14(d_i(i_2, j_2), dr_i(i_1, j_1));
  End Switch
ELSE IF (d_i(i_1, j_1) = 0 is on the right side of d_i(i_2, j_2) = 0)
  Switch (q_1, q_3) BEGIN
     CASE 1: q_1 > 0 and q_3 > 0
        Select dl_i(i_1, j_1) = 0 (=LMLT) and d_i(i_2, j_2) = 0 (=RMLH);
        Call Transformation21(dl_i(i_1, j_1), d_i(i_2, j_2));
     CASE 2: q_1 > 0 and q_3 < 0
        Select d_i(i_2, j_2) = 0 (=LMLH) and d_i(i_1, j_1) = 0 (= RMLT)
        Call Transformation22(d_i(i_2, j_2), d_i(i_1, j_1));
     CASE 3: q_1 < 0 and q_3 > 0
        /* FDH does not overlap FDT */
        Call Transformation23(d_i(i_1, j_1));
     CASE 4: q_1 < 0 and q_3 < 0
        Select d_i(i_1, j_1) = 0 (=LMLT) and dr_i(i_2, j_2) = 0 (=RMLH);
        Call Transformation24(d_i(i_1, j_1), dr_i(i_2, j_2));
  End Switch
ELSE /* the line d_i(i_1, j_1) intersects the line d_i(i_2, j_2) */
      Select d_i(i_1, j_1) = 0 and d_i(i_2, j_2) = 0;
      Call Transformation13(d_i(i_1, j_1), d_i(i_2, j_2));
END Region_Partition
```

Fig. 3. Algorithm of selecting the bounds of the transformed loop.

After selecting one or two appropriate lines, Algorithm Region_Partition executes one among eight procedures, *i.e.*, Transformation11 ~ Transformation24, which are algorithms of transforming the original loop as shown in Fig. 4. In this algorithm, the expressions $j = A_1 i + B_1$ and $j = A_2 i + B_2$ used in the index bounds correspond to the first and the second input parameter in each procedure, respectively. In Procedure Transformation11(), the first input parameter is $dl_i(i_2, j_2) = 0$ (LMLH) and the second one is $d_i(i_1, j_1) = 0$ (RMLH), where $A_1 = (1-\alpha_{11})/\beta_{11}$, $B_1 = -\gamma_{11}/\beta_{11}$, $A_2 = (1-\alpha_{21})/\beta_{21}$, and $B_2 = -\gamma_{21}/\beta_{21}$ which are derived from (5) and (6). We know that two input parameters can be the upper or lower bound in the transformed loops based on the corresponding region of the loop. The first input parameter $dl_i(i_2, j_2) = 0$ is the upper boundary in AREA3 and lower boundary in AREA1. The second one $d_i(i_1, j_1) = 0$ is the upper boundary in AREA2 and the lower boundary in AREA3. In this case, the iteration space is divided into two parallel regions, AREA1 and AREA2, and one serial region, AREA3, by the two selected lines. The execution order is AREA1 → AREA3 → AREA2.

When the input parameter is only one like Procedure Transformation12(), the iteration space is divided into two parallel regions, AREA1 and AREA2, by the selected line.

```
Procedure Transformation11(dl_i(i_2, j_2), d_i(i_1, j_1))
   /* the iteration space can be partitioned as follows */
   AREA1: {(i_2, j_2) | dl_i(i_2, j_2) > 0 },
   AREA3: {(i_2, j_2) | dl_i(i_2, j_2)   0 } ∩ {(i_1, j_1) | d_i(i_1, j_1)
0},
   AREA2: {(i_1, j_1) | d_i(i_1, j_1) < 0 }
Procedure Transformation13(d_i(i_1, j_1), d_i(i_2, j_2))
    AREA1: {(i_2, j_2) | d_i(i_2, j_2) < 0 },
     AREA3: {(i_1, j_1) | d_i(i_1, j_1)   0} ∩ {(i_2, j_2) | d_i(i_2, j_2)
0 },
     AREA2: {(i_1, j_1) | d_i(i_1, j_1) > 0 }
Procedure Transformation14(d_i(i_2, j_2), dr_i(i_1, j_1))
     AREA1: {(i_2, j_2) | d_i(i_2, j_2) > 0 },
     AREA3: {(i_2, j_2) | d_i(i_2, j_2)   0 } ∩ {(i_1, j_1) | dr_i(i_1, j_1)
0},
     AREA2: {(i_1, j_1) | dr_i(i_1, j_1) < 0 }
Procedure Transformation21(dl_i(i_1, j_1), d_i(i_2, j_2))
    AREA1: {(i_2, j_2) | d_i(i_2, j_2) < 0 },
     AREA3: {(i_2, j_2) | d_i(i_2, j_2)   0 } ∩ {(i_1, j_1) | dl_i(i_1, j_1)
0},
     AREA2: {(i_1, j_1) | dl_i(i_1, j_1) > 0 }
Procedure Transformation22(d_i(i_2, j_2), d_i(i_1, j_1))
    AREA1: {(i_2, j_2) | d_i(i_2, j_2) > 0 },
     AREA3: {(i_2, j_2) | d_i(i_2, j_2)   0 } ∩ {(i_1, j_1) | d_i(i_1, j_1)
0},
     AREA2: {(i_1, j_1) | d_i(i_1, j_1) < 0 }
Procedure Transformation24(d_i(i_1, j_1), dr_i(i_2, j_2))
    AREA1: {(i_2, j_2) | dr_i(i_2, j_2) < 0 },
     AREA3: {(i_2, j_2) | dr_i(i_2, j_2)   0 } ∩ {(i_1, j_1) | d_i(i_1, j_1)
0},
     AREA2: {(i_1, j_1) | d_i(i_1, j_1) > 0 }
Procedure Transformation12(d_i(i_1, j_1))
    AREA1: {(i_1, j_1) | d_i(i_1, j_1)   0 }
```

```
    AREA2: {(i1, j1) | di(i1, j1) < 0 }
Procedure Transformation23(di(i1, j1))
   AREA1: {(i1, j1) | di(i1, j1)   0 }
    AREA2: {(i1, j1) | di(i1, j1) > 0 }
```

Fig. 4. Algorithms of transforming the original loop.

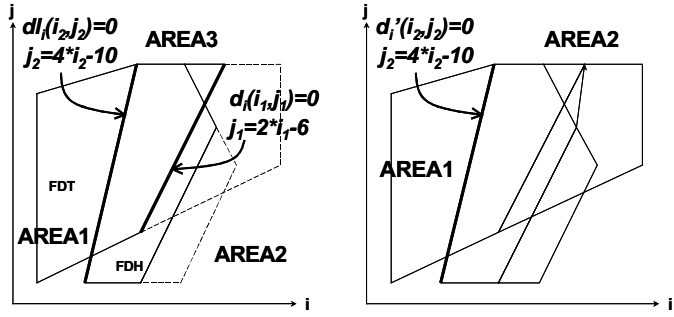


Fig. 5. Regions of the loop partitioned by (a) the improved region partitioning, (b) the unique set oriented partitioning in Example 1

Fig. 5(a) shows regions of the loop partitioned by our proposed technique in Example1.

In this case, the iteration space is divided into two parallel regions, AREA1 and AREA2, and one serial region, AREA3, by the two selected lines $j = 4i_2 - 10$ and $j = 2i_1 - 6$. The execution order is AREA1 → AREA3 → AREA2. Transformed loops are given as follows.

```
/* AREA1 – parallel region */
doall i = l1, u1
    doall j = max(l2, 4i2 - 10), u2
       A(2*j+3, i+j+5) = . . .
                . . .  =A(2*i+j-1, 3*i-1)
    enddoall
enddoall
/* AREA3 – serial region */
do i = l1, u1
   do j = max(l2, 2i1 - 6), min(u2, 4i2 - 10)
       A(2*j+3, i+j+5) = . . .
                . . .  =A(2*i+j-1, 3*i-1)
    enddo
enddo
/* AREA2 – parallel region */
doall i = l1, u1
   doall j = l2, min(u2, 2i1 - 6 )
       A(2*j+3, i+j+5) = . . .
```

```
                . . .  =A(2*i+j-1, 3*i-1)
    enddoall
  enddoall
```

Fig. 6. Transformation of the loop by the improved region partitioning method in Example 1.

4 Performance Analysis

Theoretical speedup for performance analysis can be computed as follows. Ignoring the synchronization, scheduling and variable renaming overheads, and assuming an unlimited number of processors, each partition can be executed in one time step. Hence, the total time of execution is equal to the number of parallel regions, N_p, plus the number of sequential iterations, N_s. Generally, speedup is represented by the ratio of total sequential execution time to the execution time on parallel computer system as follows:

$$Speedup = (N_i * N_j)/(N_p + N_s)$$

where N_i, N_j are the size of loop i, j, respectively

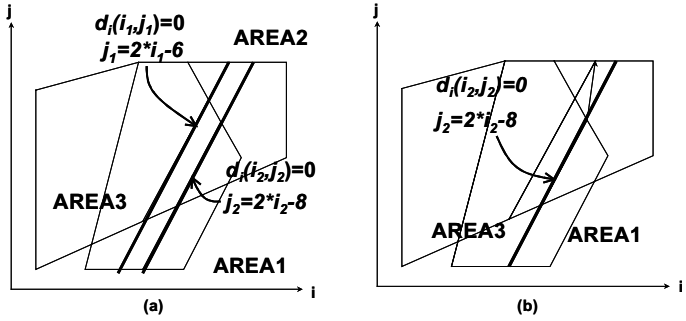


Fig. 7. Regions of the loop partitioned by (a) the improved three region partitioning method, (b) the three region partitioning method in Example 1

By using an example given in Example 1, we compare the performance of our proposed method with that of related works. The improved three region partitioning [13] can divide the iteration space into two parallel regions, AREA1 and AREA2, and one serial region, AREA3, by line $d_i(i_2, j_2) = 0$ ($j_2 = 2i_2 - 8$) and line $d_i(i_1, j_1) = 0$ ($j_1 = 2i_1 - 6$) as shown in Fig. 7(a). The speedup can be computed as (10*10)/(2+55) = 1.75.

The three region partitioning [11] is similar to the improved three region partitioning. This method divides the iteration space into one parallel region, AREA1, and one serial region, AREA3, by line $d_i(i_2, j_2) = 0$ ($j_2 = 2i_2 - 8$) as shown in Fig. 7(b). The speedup can be computed as (10*10)/(2+66) = 1.47.

Applying the unique set oriented partitioning to this loop illustrates case 4 of [10]. The line $j_2 = 4i_2 - 10$ as shown in Fig. 5(b) can divide the iteration space into two

parts: AREA1 is the left side of this line, and contains only flow dependence tails, and AREA2 is the right side of this line, and is the serial region. AREA2 is tiled into 34 tiles with width = 1 and height = 2, thus the speedup for this method is (10*10)/(1+34) = 2.9.

Applying the proposed method to this loop is the case that FDT overlaps with FDH and line $d_i(i_1, j_1) = 0$ does not intersect line $d_i(i_2, j_2) = 0$. After variable renaming, there remains only flow dependence as shown by the partitioning in Fig. 5(a). Thus, lines LMLH, $dl_i(i_2, j_2) = 0$ ($j_2 = 4i_2 - 10$), and $d_i(i_1, j_1) = 0$ ($j_1 = 2i_1 - 6$) divide the iteration space into two parallel regions, AREA1 and AREA2, and a serial region, AREA3. AREA1, the left side of line $dl_i(i_2, j_2) = 0$ ($j_2 = 4i_2 - 10$), is only flow dependence tail set. AREA3 between the right side of line $dl_i(i_2, j_2) = 0$ ($j_2 = 4i_2 - 10$) and the left side of line $d_i(i_1, j_1) = 0$ ($j_1 = 2i_1 - 6$) contains both flow dependence head and tail set. AREA2, the right side of line $d_i(i_1, j_1) = 0$ ($j_1 = 2i_1 - 6$), is only flow dependence head set. Fig. 6 shows transformation of the loop by our proposed method in Example 1. The speedup for this method is (10*10)/(2+18) = 5.

In the above comparisons, our proposed partitioning method exploits more parallelism than the other related methods.

5 Conclusions

In this paper, we studied the problem of transforming nested loops with non-uniform dependences to maximize parallelism, and proposed a new approach to it, the Improved Region Partitioning method based on Convex Hull theory. The method can divide the iteration space into two parallel regions and one or less serial region after variable renaming.

IF FDT (Flow Dependence Tail set) does not overlap FDH (Flow Dependence Head set), a line $d_i(i, j) = 0$ between two sets divides the iteration space into two areas. The iterations within each area can be fully executed in parallel.

In case which FDT overlaps FDH, two selected lines divide the iteration space into two parallel regions as large as possible and one serial region as small as possible.

In comparison with some previous partitioning methods, such as minimum dependence distance tiling, unique sets oriented partitioning and three region partitioning, the proposed method gives much better speedup and extracts more parallelism than other methods.

Our future research work is to develop a method for improving parallelization of higher dimensional nested loops

References

1. D. Kuck, A. Sameh, R. Cytron, A.V.C. Polychronopoulos, G. Lee, T. McDaniel, B. Leasure, C. Beckman, J. Davies, and C. Kruskal, "The effects of program restructuring, algorithm change and architecture choice on program performance," in Proceedings of the 1984 International Conference on Parallel Processing, (1984)
2. Y. S. Chen, S. D. Wang, and C. M. Wang, "Tiling nested loops into maximal rectangular blocks," Journal of Parallel and Distributed Computing, vol. 35, no.2, (1996) 123-132

3. M. Wolfe, "Loop skewing: The wavefront method revisited," International Journal of Parallel Programming, (1986) 279-293
4. U. Banerjee, Loop Transformations for Restructuring compilers. Kluwer Academic Publishers, (1993)
5. E. Wolfe and M. S. Lam, "A loop transformation theory and an algorithm to maximize parallelism," IEEE transactions on Parallel and Distributed Systems, vol. 2, (1991) 452-471
6. U. Banerjee, Dependence Analysis for Supercomputing, Kluwer Academic, Norwell, Mass., (1988)
7. Z. Shen, Z. Li, and P. Yew, "An empirical study on array subscripts and data dependences," in Proc. Int. Conf. Parallel Processing, vol. II, (1989) 145-152
8. T. Tzen and L. Ni, "Dependence uniformization: A loop parallelization technique," IEEE Trans. Parallel and Distributed Systems, vol. 4, no. 5, (1993) 547-558
9. S. Punyamurtula and V. Chaudhary, "Minimum dependence distance tiling of nested loops with non-uniform dependences," in Proc. Symp. Parallel and Distributed Processing, (1994) 74-81
10. J. Ju and V. Chaudhary, "Unique sets oriented Partitioning of nested loops with non-uniform dependences," in Proc. Int. Conf. Parallel Processing, vol. III, (1996) 45-52
11. A. Zaafrani and M. R. Ito, "Parallel region execution of loops with irregular dependencies," in Proc. Int. Conf. Parallel Processing, vol. II, (1994) 11-19
12. M. Wolfe and C. W. Tseng, "The power test for data dependence," IEEE Trans. Parallel and Distributed Systems, vol. 3, no. 5, (1992) 591-601
13. C. K. Cho and M. H. Lee, "A Loop Parallization Method for Nested Loops with Non-uniform Dependences", in Proceedings of the International Conference on Parallel and Distributed Systems, (1997) 314-321

Fair Exchange to Achieve Atomicity in Payments of High Amounts Using Electronic Cash

Magdalena Payeras-Capella, Josep Lluís Ferrer-Gomila, and
Llorenç Huguet-Rotger

Departament de Ciències Matemàtiques i Informàtica.
Universitat de les Illes Balears.
Carretera de Valldemossa, Km. 7.5, 07122 Palma de Mallorca
{mpayeras, dijjfg, dmilhr0}@uib.es

Abstract. Payments involving high amounts of money must be extremely secure. Security aspects have to be implemented in the payment system (in this case electronic coins) and in the fair exchange between the coin and the product. A secure payment scheme for high amounts has to provide prevention of forgery, overspending, double spending and robbery. This paper describes a fair exchange protocol that can be used together with a payment system for high amounts. This way, the fair exchange of elements represents the addition of atomicity to the payment system.

1 Introduction

Among the presented electronic payment systems, there are, in one hand, electronic cash payment systems, and among them, protocols for payments of high amounts that achieve total fraud prevention (counterfeiting, robbery and double spending) to assure the validity of payments. In the other hand, there are protocols that allow a fair exchange between the electronic coin used in the payment and the purchased product or its receipt [3, 4, 7, 11].

The purchase of products with high prices requires both total fraud prevention in the use of coins and fair exchange between product and electronic coin. This fact leads to the definition of a specific protocol for this kind of transactions: a fair payment protocol. The fair exchange protocol for the payments of high amounts is based in the use of an electronic payment protocol that provides the required security for the payments of high amounts, like the protocol described in [5]. The exchange protocol will maintain the security and anonymity features of the used coins, and will achieve the desired features of the exchange. These features are: bilateral certified delivery and involvement of the TTP only for conflict resolution.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 223–232, 2004.
© Springer-Verlag Berlin Heidelberg 2004

2 Fair Exchange between Electronic Coins and Products (or Purchase Receipts)

The fair exchange of a coin with a receipt o with a product is useful when electronic cash systems are used for the purchase of goods or services offered electronically. In payments using credit cards, the purchase order including the card number (and the signature) is exchanged for a receipt or a digital good. We consider that the payment using electronic cash requires a specific exchange protocol. The payment using credit card can be considerer an application of the contract signing protocols [6]. In the payments using electronic cash, the coins are part of the exchange (item provided by the buyer). These exchanges cannot be considered an application of the contract signing protocols due to the specific situations in where the interruption of the exchange can derive in the loss of a coin for both parts or the loss of anonymity of an anonymous user. As an example, in an off-line anonymous system, if the buyer does not know if the seller has received the coin, he cannot spend the coin again, because if the seller has received the coin the buyer would be identified and accused of reutilization of coins.

Interruptions can be due to net failures or fraudulent behavior of any part. Consequently, is possible that the buyer provides the coin and do not obtain the good or receipt from the seller, or that the seller sends the good and do not receive the coin. Atomicity allows linking a group of operations so they must be executed totally or not executed at all.

An atomic interchange is fair for both parts. Moreover, is interesting that each part can prove which element the other part has received. With this feature, in case of dispute, each part can provide proof of the conclusion of the exchange.

In [12], *money atomicity* is defined as the feature that avoids the creation or destruction of money during the exchange (although this is not a fair exchange). [12] also defines *good atomicity.* This definition describes the protocols that provide *money atomicity* and the fair exchange of good and coin.

Bilateral certified delivery [12] presents together *good* and *money atomicity,* and gives (to both parts) proof of the elements that the other part has received. With *unilateral certified delivery* [3], the buyer can prove which goods received, in case of dispute, like when the received goods do not fulfill the description. The seller only receives the payment if the buyer receives the good or receipt, and cannot prove if the buyer received the good.

Atomic certified delivery [9] provides good and money atomicity and the parts have agreed in an initial negotiation and the exchange gives proof of reception. Finally, *distributed certified purchase* [9] presents good and money atomicity when more than one seller is involved in the purchase.

2.1 Objectives

For this application, the objective is an atomic and certified purchase using electronic payment. This way, a seller would be able to prove that the buyer has received the product (or receipt) and the buyer would be able to prove that the seller has received the coin and to prove which is the received product.

Anonymity is a desired feature, at least the anonymity of the buyer (as in some electronic cash systems). Another ideal feature is the execution of the exchange protocol without the involvement of a TTP. In an optimistic protocol, the TTP only is necessary if the exchange cannot be completed. For the same reason, is desirable that the payment system does not require a bank validation during the payment (off-line payment).

2.2 Previous Work

Existing solutions can be grouped in function of the role of the TTP. In the solution presented in [11], a coordinator knows the identity of all parts, so the payments cannot be anonymous. This solution is useful in case of network failures but is not useful in case of fraud intent attempt.

Some solutions, like [4] do not require a TTP. In this case, the coin is divided in two parts that are sent before and after the reception of the product. The seller is not protected because cannot request the participation of a TTP in case the second part of the coin is not received. The coins can be in an ambiguous estate if the buyer does not want to assume the risk of being identified. As a conclusion, the scheme does not provide atomicity, only a certain degree of protection of the buyer.

Other protocols execute the exchanges with the participation of the TTP, like [3, 7, 10, 12]. In [7] the TTP is active (a *blackboard*) where all users can read and write. In [3] the bank, which acts as a TTP, is involved in the payment. The scheme provides unilateral certified delivery. [12] presents an *on-line* payment where the bank acts as a TTP and guarantees the fair exchange during the payment. Other similar schemes are [10] and [9] where an *on-line* payment coordinator is used.

Other schemes use passive TTP (optimistic exchange), but do not achieve the ideal features for the atomic payment. In [15], if the exchange does not conclude satisfactorily, the buyer can obtain the coin (payment cancellation), but is not able to finish the exchange.

Other solution is [13]. This paper does not specify the payment system used for the exchange. The purchase is not certified, the seller cannot prove that the client has received the product.

3 Payment of High Amounts: Protocol Description

The protocol described in [5] is an electronic payment scheme that allows the prevention of double spending without using tamper resistant devices and using an off-line payment capture. The scheme is protected against fraud, counterfeiting, overspending, double spending and robbery. The scheme allows anonymous and untraceable payments. Moreover, the receiver of the payment can use the coins in an anonymous way. These features are suitable for the payments of high amounts.

The scheme includes *withdrawal, transfer* and *deposit* sub-protocols. The term *e-note* is used to refer the electronic bank notes. In the description A is the payer, B is the payee and F is the bank. Other notation and elements are as follows:

ID_X	identity of actor *X*
Q_i	amount to be withdrawn, transferred or deposited
Y, Z	concatenation of two messages (or tokens) *Y* and *Z*
$Sign_i(Y)$	digital signature of principal *i* on message *Y*
$i \rightarrow j$: Y	principal *i* sends message (or token) *Y* to principal *j*
$E_K(M)$	symmetric encryption of message *M* with key *K*
$PU_F(K)$	key *K* enciphered with the public key of the bank
$SN_i=H(SP_i)$	serial number, *SN* of an *e-note*, hash of a random secret proof, *SP*
$M_i=Sign_{Fx}(SN_i)$	signature on SN_i with a private key indicating value *x*

In the *withdrawal* sub-protocol, a merchant, *A*, requests an *e-note* to *F*. *F* creates an *e-note* and debits *A*'s account:

1. $A \rightarrow F$: $ID_A, Q_1, SN_1, Sign_A(Q_1, SN_1)$
2. $F \rightarrow A$: Q_1, SN_1, M_1

A generates a random number SP_1 (the secret proof to validate the *e-note*), that must be kept secret. SN_1 (the serial number of the future *e-note*) is the result of a hash function applied to SP_1. *A* proves ownership of his account signing the serial number and the amount, Q_1. *F*'s signature on SN_1, together with SP_1, is the *e-note*. SN_1 will be used to prevent double spending of the *e-note*. *A* can prove the ownership with the knowledge of SP_1 and M_1.

To redeem an *e-note*, the owner must show the knowledge of the *e-note* secret proof (SP_1), but he is not forced to reveal his identity. If *F* saves all available information about the *e-note*, it could recognize that *e-note* at deposit, but thanks to the use of the *transfer* sub-protocol (see section 2.2), the bank (or the collusion of the bank and the merchant depositing the *e-note*) cannot reveal where *A* spent it. Therefore, payments will be anonymous and untraceable, and the scheme is secure against money forging.

When *A* wants to pay to *B*, *A* executes the following *transfer* sub-protocol:

1. $A \rightarrow B$: Purchase_order
2. $B \rightarrow A$: $Q_i, SN_i, Sign_B(Q_i, SN_i)$
3. $A \rightarrow F$: $PU_F(K), E_K(SN_j, M_j, SP_j, Q_i, SN_i, SN_r)$
4. $F \rightarrow A$: M_i, M_r
5. $A \rightarrow B$: SN_i, M_i

B sends to *A* a serial number (SN_i), the price of the item (Q_i) and the digital signature of the previous information, as a response to the *Purchase_order*, without revealing the secret proof (SP_i). *A* will request to *F* an *e-note* to pay *B*, with the serial number given by *B* (SN_i). *A* sends her *e-note* M_j to the bank, with the associated secret

proof (SP_j). The request is encrypted with a session key (K), so nobody can intercept SP_j. A indicates the amount (Q_i) of the *e-note* M_j to be converted in the new *e-note* using SN_i. The remaining fraction Q_r (if $Q_i < Q_j$) will be used to create another *e-note* with serial number SN_r. F cannot find out the identities of the users.

If SN_j is found in the list of spent *e-notes*, F has detected a double spending attempt, and will abort the operation. If the *e-note* (SP_j, M_j) is valid, F creates the new *e-notes* M_i and M_r, and sends them to A. A knows the *e-note* M_i and SN_i, but A does not know SP_i. A stores the information related to the payment during an established period. This information can be requested in case of anonymity revocation. The scheme is anonymous for A, because B does not know the identity of A.

B checks the validity of the *e-note* M_i (verifying the signature of F). Only B knows SP_i and he is the only one that can spend that *e-note*. He does not need to contact F. Now, B has an *e-note* with the same properties that a withdrawn one. B can deposit it identifying his account. In addition, B can use the *e-note* for a new payment, but a collusion between A and F will be able to trace B. To solve this problem B has to use the auto-transfer sub-protocol.

Transfer Sub-protocol Applied to Auto-transfer. A knows SN_i and B's identity. So, payments with that *e-note* could be traced by the collusion of A and F. The solution is the auto-transfer operation:

1. $B \rightarrow F$: $PU_F(K), E_K(SN_i, M_i, SP_i, Q_s, SN_s, SN_t)$
2. $F \rightarrow B$: M_s, M_t

B calculates SN_s and SN_t from the random secret proofs SP_s and SP_t, respectively. B requests F that a specific *e-note* is going to be transferred. B sends SP_i encrypted with a session key (and other information analogous to the previous case).

If the *e-note* is valid (e.g., not double spent), F creates two new *e-notes* with the new serial numbers and the required values, and SP_i is appended to the list of spent *e-notes*. F does not know who is the user auto-transferring the *e-note*. Furthermore, F cannot distinguish if the user is auto-transferring the total amount of the *e-note*, if he is preparing a payment with a fraction of the *e-note* and auto-transferring the remaining part, or if he is preparing two payments. The auto-transfer subprotocol can also be used by A before the payment.

In the *deposit* sub-protocol, it is necessary an identification of the merchant's account:

1. $B \rightarrow F$: $PU_F(K), E_K(ID_B, SN_i, M_i, SP_i, Q_i), Sign_B(ID_B, SN_i, M_i, SP_i, Q_i)$

B sends the secret proof SP_i, and some identifying information (to deposit money in the right account), all encrypted with a session key K. F checks the validity of the *e-note*, and if it is correct then credits B's account.

The protocol achieves security requirements: *e-notes* cannot be counterfeited (thanks to the use of the bank private keys), overspending and double spending are avoided (*e-notes* are created after a debit in an user account, and the bank saves the list of redeemed serial numbers, deposited and transferred, with their secret proofs), and

stolen *e-notes* cannot be redeemed (it is necessary the secret proof, and it is encrypted when is transmitted). On the other hand, the scheme provides anonymity and untraceability to payers and payees, thanks to the *auto-transfer* subprotocol. *E-notes* can be transferred multiple times without depositing and without any identification. Payments between the same pair of merchants are unlinkable. There is not any relationship between them: new serial numbers are used in each payment.

The scheme prevents illegal activities (as blackmailing, money laundering and illegal purchases/sales). For example, if blackmailing is suspected or reported, the appropriate authority will allow to the bank to demand the identity of the user who will try to *transfer* or *deposit* the suspicious serial number (*SN*). We do not use blind signatures to achieve anonymity, and so a blackmailed user always knows the serial number of the money given to the blackmailer. If money laundering is suspected, the authority will allow to the bank to demand user identification when this user is going to transfer the money.

The possibility of anonymous payment and redemption, the double spending prevention and other security properties, makes this scheme suitable for anonymous payments of high amounts in B2B transactions. However, atomicity is not achieved in this payment system, so a fair exchange protocol is required for atomic purchases.

4 Description of the Fair Exchange Protocol

The fair exchange protocol uses an *interchange* subprotocol executed each time that a purchase requires the payment of high amounts. This subprotocol does not require the intervention of a TTP and allows finalizing the atomic purchase (including the payment).

When the *interchange* subprotocol cannot be finalized, both parts can request to the TTP the finalization of the exchange using the *finish* subprotocol. This protocol does not require a *cancel* subprotocol due to the features of the used payment system. Due to [5] does not allow double spending, a challenge-response stage (to identify double spenders) is not required, and the number of interactions required in the fair payment is lower than in other schemes.

4.1 Interchange Subprotocol

The *interchange* subprotocol is formed by an initial step (that contains the selected product or service) and the three steps explained below:

- Step 1: The seller sends the encrypted product or receipt to the buyer.
- Step 2: The buyer sends the coin to the seller.
- Step 3: The seller sends the encryption key to the buyer.

The coin used in the payment is the *e-note* described in section 3 without any modification, $coin=\{Sign_{Fx}(SN_i)\}$, where SN_i is the identifier of the coin and SP_i is the related secret key known only by the receiver.

The *interchange* subprotocol is as follows:

0. A→B:	Purchase_order	*First step of the transfer subprotocol*
1. B→A:	SN_i, Q_i, $Sign_B(Q_i, SN_i)$ $c=E_k$(product/receipt) $K_i=PU_T(k)$ $H_M=PR_M\{H[H(c), k_T, SN_i]\}$	*Modified 2nd step of the transfer subp.*
2. A → B:	M_i	*Step 5 of the transfer subprotocol*
3. B → A:	K	*New step*

With the modification of step 2 and the incorporation of the new step, the whole protocol for fair payment is as follows:

A → B:	Purchase_order
B → A:	SN_i, Q_i, $Sign_B(Q_i, SN_i)$; $c=E_k$(product/receipt) $K_i=PU_T(k)$; $H_M=PR_M\{H[H(c), k_T, SN_i]\}$
A → F:	$PU_F(K_2)$, $E_{K2}(SN_j, M_j, SP_j, Q_i, SN_i, SN_r)$
F → A:	M_i, M_r
A → B:	M_i
B → A:	K

4.2 Finish Subprotocol

If the *interchange* subprotocol does not conclude, then a part can be mistreated in front of the other. The interruption of the exchange can be due to failure or misbehavior, so the other part has to be able to return the exchange to a fair situation.

In this protocol, the interruption can be produced after the reception of the message of step 1 or after the message of step 2. In the first case, the parts do not have the desired element, and any compromising element has been sent, so a *cancel* subprotocol is not required.

The *finish* subprotocol allows the parts to obtain the desired element from the TTP. The *finish* subprotocol uses two boolean variables: *finished* and *proved,* that show if *A* has sent the coin and as a consequence has received the key, and that *B* have provided the secret proof, respectively. The default value of both variables is *false.* Both parts can execute the finish subprotocol.

B → T: Request, SN_i, Q_i, E_k(product/receipt), k_T, h_M

T: **IF** (*finished* = *true*)

T → B: M_i

ELSE

T → B: Request of the Secret Proof

B → T: SP_i

T: *Proved = true*

A → T: Request, $Sign_B(SN_i)$, Q_i, M_i, E_k(product/receipt), k_T, h_M

T → A: K

T: *Finished = true*

IF (*proved = true)*

T → F: M_i, SP_i, Deposit request (ID_B)

B will execute the subprotocol if he does not receive the message of step 2. In this case, *A* will not receive the message of step 3, so *A* can also execute the *finish* subprotocol. If *A* does not receive the message of step 3 once *B* has received the message of step 2 and has all the elements of the payment, *A* can execute the *finish* subprotocol to obtain the encryption key.

5 Evaluation

The fair purchase protocol can be concluded executing exclusively the *interchange* subprotocol o executing both *interchange* and *finish* subprotocols. The states associated with these executions are:

- *A* does not send the message of step 2.
  If *A* decides not to conclude the payment (once the purchase_order has been sent), he will not request the coin to *F*. Consequently, *A* will not execute the *finish* subprotocol. If *B* executes the *finish* subprotocol, *T* will check if *A* has executed the *finish* subprotocol previously (*finish = true)*. *T* will not send the coin to *B* if the coin has not been created because in this case *A* cannot execute the *finish* subprotocol.

- *A* sends the message of step 2 and *B* does not receive it.
  A is hoping to receive the key and *B* is hoping to receive the coin. Both parts can execute the *finish* subprotocol. These executions can be made in different order.
    - First *A* executes *finish*, then *B* executes *finish*

A sends the coin to *T* and receives the key, then *finished* is *true. B* will obtain the coin.

- First *B* executes *finish*, then *A* executes *finish.*
  B executes the *finish* subprotocol, and *finished* is *false. B* receives a message requesting the coin secret proof (SP_i), which will be useful if later *A* executes the *finish* subprotocol. When *A* executes the *finish* subprotocol, the variable *finished* is *true.* For this reason *A* will obtain the key and *T* will deposit the coin in *B*'s account.

- *A* sends the message of step 2, *B* receives it and try to cheat (*B* does not send the message of step 3).
  B cannot prevent that *A*, once the coin has been created, obtains the encryption key if he sends the coin to the TTP, executing the *finish* subprotocol. In this case, *B* has the coin, so this is a fair situation.
- *B* sends the message of step 3 and *A* does not receive it.
  This case is equivalent to the previous one.

6 Conclusions

The electronic payments representing the transfer of high amounts require very secure protocols. The purchase of expensive goods and services can be considered an exchange between the product or receipt and the payment. It is necessary to use a secure payment system that guarantees the validity of the coins, but also the seller has to be sure that he will receive the payment if he provides the product.

From the buyer point of view, is important that the purchase represents a fair exchange, the buyer would be unsure if the system does not assure him that he will receive the product or receipt if he executes the payment involving a high amount. The need of atomicity is especially present in payments of high amounts.

The security aspects related with the coin are satisfied due to the use of a specific payment system for high amounts, like [5].

The fair purchase can be done using the *interchange* subprotocol, which is a set of steps of the payment protocol lightly modified, forming a three steps exchange protocol. A *finish* subprotocol, executed between one of the parts and the trusted third party is used when the exchange produces an unfair situation. Due to a reduced number of transfers, the possibilities of interruption of the *interchange* protocol are also reduced. In all these situations, the use of the *finish* subprotocol allows the parts to achieve fairness.

References

[1] Adi, K., Debbadi, M. and Mejri, M.: "A new logic for electronic commerce protocolos.", AMAST'00, LNCS 1816, pages 499-513, Springer Verlag, 2000.

[2] Asokan, N., Herreweghen, E. Van. and Steiner, M.: "Towards a framework for handling disputes in payment systems", 3rd Usenix workshop on electronic commerce, pages 187-202, 1998.

[3] Camp, J., Harkavy, M., Tygar, J.D. and Yee, B.: "Anonymous atomic transactions", 2nd USENIX workshop on electronic commerce, pages 123-133, 1996.

[4] Jakobsson, M.: "Ripping coins for a fair exchange", Eurocrypt'95, LNCS 921, pages 220-230, Springer Verlag, 1995.
[5] Ferrer, J.L., Payeras, M. and Huguet, L.: "A fully anonymous electronic payment scheme for B2B", International Conference on Web Engineering, 2003, ICWE'03 proceedings, LNCS 2722, pages 76-79, Springer Verlag, 2003.
[6] Ferrer, J.L., Payeras, M. and Huguet, L.: "Efficient optimistic *N*-Party contract signing protocol", Information Security Conference. 4th International Conference, ISC'01, LNCS 2200, pages 394-407, Springer Verlag, 2001.
[7] Pagnia, H. and Jansen, R.: "Towards multiple payment schemes for digital money", Financial Cryptography' 97, LNCS 1318, pages 203-216, Springer Verlag, 1997.
[8] Schuldt, H., Popovivi, A. and Schek, H.: "Give me all I pay for – Execution guarantees in electronic commerce payment processes", Informatik'99 –Workshop "Unternehmensweite und unternehmensübergreifende Workflows: Konzepte, Systeme, Anwendungen", 1999.
[9] Schuldt, H., Popovivi, A. and Schek, H.: "Execution guarantees in electronic commerce payments.", 8th international workshop on foundations of models and languages for data and objects (TDD'99), LNCS 1773, Springer Verlag, 1999.
[10] Su, J. and Tygar, J.D.: "Building blocs for atomicity in electronic commerce", 6th USENIX security symposium, 1996.
[11] Tang, L.: "Verifiable transaction atomicity for electronic payment protocols", IEEE ICDCS'96, pages 261-269, 1996.
[12] Tygar, J.D.: "Atomicity in electronic commerce", 15th annual ACM symposium on principles of distributed computing", pages 8-26, 1996.
[13] Vogt, H. and Pagnia, H. and Gärtner, F.C.: "Modular fair exchange protocols for electronic commerce" 15th Annual Computer Security Applications Conference, ACSAC'99, pages 3-11, 1999.
[14] Wong, H.C.: "Protecting individuals' interests in electronic commerce protocols", Ph. D. Thesis. Carnegie Mellon University.
[15] Xu, S., Yung, M., Zhang, G. and Zhu, H. "Money conservation via atomicity in fair off-line e-cash", International security workshop ISW'99, LNCS 1729, pages 14-31, Springer Verlag, 1999.

Gossip Based Causal Order Broadcast Algorithm

ChaYoung Kim[1], JinHo Ahn[2⋆], and ChongSun Hwang[1]

[1] Dept. of Computer Science & Engineering, Korea University
5-1 Anamdong, Sungbukgu, Seoul 136-701, Republic of Korea
{chayoung,hwang}@disys.korea.ac.kr
[2] Dept. of Computer Science, College of Information Science, Kyonggi University
San 94-6 Yiuidong, YeongTonggu, Suwon Kyonggido 443-760, Republic of Korea
jhahn@kyonggi.ac.kr

Abstract. Reliable group communication facility with two message ordering constraints, atomic ordering and causal ordering, is essential for distributed systems. But, as the system size rapidly increases, traditional group communication algorithms become unsuitable for very large-scale systems due to their strong reliability properties. To solve the problem, several gossip-based algorithms were presented to significantly improve scalability by ensuring the reasonably weak reliability condition. They are all designed to guarantee totally ordered delivery. However, many distributed applications such as multimedia systems and collaborative work, require causally-ordered message delivery. In this paper, we propose a Probabilistic Causal order BroadCast algorithm, *PCBCast*, to preserve the inherent scalability of the gossip style approach compared with the existing ones.

Keywords: Group communication, causal order delivery, gossip, scalability, reliability

1 Introduction

In an asynchronous distributed system, members of a process group generally use group communication facilities in order to cooperate with each other to perform a distributed computation[5]. A broadcast protocol orders messages relative to failure and recovery events, instead of an agreement protocol which could be slow because of its synchronization overhead. In particular, when inconsistencies may arise in case that messages are delayed in transit, reliable group communication mechanisms could resolve this problem in the system by imposing delivery ordering constraints and all or nothing properties[1]. There are two representative message ordering semantics, atomic ordering and causal ordering, in reliable group communication of distributed systems. Atomic ordering broadcast is useful for building voting-style protocols, because it is guaranteeing all messages are delivered in a consistent total order. Causal ordering broadcast ensures that if two messages are causally related and have the same destination, they are

⋆ Corresponding author. Tel.:+82-31-249-9674; fax:+82-31-249-9673.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 233–242, 2004.
© Springer-Verlag Berlin Heidelberg 2004

delivered to the application in their sending order. It is more lightweight than atomic one by involving minimal delivery latency[1,2].

Traditional reliable group communication protocols are not scalable in very large-scale systems like Internet, due to their strong reliability properties[3,4,7,8, 10,11,14,16,20]. To solve the problem, several broadcast protocols were proposed based on peer to peer interaction models for offering scalability[18]. First, techniques such as SRM(Scalable Reliable Multicast Protocol)[9] or RMTP (Reliable Message Transport Protocol)[19] include scalable multicast protocols which overcome message losses or failures and focus upon best-effort reliability in IP network layer. However, tracking membership remains an issue in network-level multicast approaches and the lack of wide deployment of IP multicast limits their applicability. As a result, some application-level multicast protocols like SCRIBE[4] and CAN-multicast[21] have recently emerged as attractive alternatives. Probabilistic versions of these have received much attention and provide good scalability and reliability properties[3], but require the existence of such large-scale peer-to-peer routing infrastructures[10,11].

In contrast, gossip-based protocols scale well to large groups, are easy to deploy and degrade gracefully as the rate of node failures or message losses increases while they do not need these infrastructures[16]. The protocols rely on epidemic style dissemination and provide probabilistic guarantees of delivery by weakening the strong reliability properties. Their scalability relies on a peer to peer interaction model because the load is distributed among all participating nodes. Also, to achieve reasonably high reliability, they use redundant messages. In these protocols, when a node generates a message, it sends the message to a randomly chosen small subset of other nodes. When a node receives the message for the first time, it does the same. The number of gossip targets is called the fanout which relates reliability of gossip-based protocols to key system parameters[7, 16]. Many variants of gossip-based protocols exist[15], which differ in the number and choice of gossip targets[11]. But, there is no existing gossip-based protocol offering causally ordered delivery property more lightweight than totally ordered delivery one. In particular, the former property is essential for many distributed applications, such as video-conferencing, multi-party games and private chat rooms[5,20]. In this paper, we propose a gossip-based broadcast algorithm guaranteeing reliable causal ordering delivery to preserve the inherent scalability of peer-to-peer interaction models.

The remainder of this paper is structured as follows: Section 2 describes the system model. In section 3, we present our algorithm that solves the probabilistic causal ordering delivery broadcast problem. Sections 4 and 5 discuss related works and conclude the paper respectively.

2 System Model

We consider a system composed of a finite set of processes P = {p1, ... , pn} that communicate only by exchanging messages over a fully connected, point-to-point network. The processes have unique, totally ordered identifiers,

and can toss weighted, independent random coins. Runs of the system proceed in a sequence of rounds in which messages sent in the current round are delivered in the next. There are two types of failures, both probabilistic in nature. The first are process failures. There is an independent, per-process probability of at most γ that a process has a crash failure during the finite duration of a protocol. Such processes are called faulty. And processes that never fails are correct. The second type of failures are message omission failures. There is an independent, per-message probability of at most δ that a message between non-faulty processes experiences a send omission failure. The union of all message omission failure events and process failure events are mutually independent. There are no malicious faults, spurious messages, or corruption of messages (i.e., we do not consider Byzantine failures.). For simplicity, we do not include process recovery in the model. We expect that both γ and δ are small probabilities. Processes communicate using the primitives send(m) and receive(m). Communication links are fair-lossy as follows:

- If p sends message m to a correct process q an infinite number of times, q receives m from p an infinite number of times.
- If p sends m to q a finite number of times, q receives m from p a finite number of times.
- If q receives m from p at time t, p sent m to q before t.

Even though fair-lossy links can lose messages, correct processes can construct reliable communication links on top of fair-lossy links by periodically retransmitting messages. If a correct process p keeps sending a message m to another correct process q, then q eventually receives m from p.

3 Gossip-Based Causal Order Broadcast Algorithm

3.1 Basic Idea

Processes executing our Probabilistic Causal order BroadCast algorithm (PCBCast) proceed in a sequence of rounds. Each process starts in round R, which is the fixed number of rounds of gossip to run. R is assigned to each message with a vector time stamp which counts the number of messages that causally precede the message. R is decremented by one. After the fixed number of rounds R becomes 0, the corresponding broadcast message is garbage-collected from the system. If process p has a broadcast message in round R and attempts to broadcast another message, it does not wait until the maximum rounds of the former has terminated and continuously broadcasts the latter. Messages are delivered immediately to the application layer at destinations in their vector time stamp order. A process may delay delivery of a broadcast message and buffer it until any messages with a vector time stamp of less than its one are delivered. But, PCBCast involves minimal delivery latency only by ensuring message causality. When concurrent messages are received or the causality is never violated, PCBCast algorithm never delays messages. After the initial reception of a message,

processes set a new time by adding its sending time and the maximum delivery delay. Then, these processes will continue to gossip about the message for R rounds. This number of rounds and the number of processes to which processes gossips in each round are important parameters to the algorithm. Like in [3], PCBCast assumes that each process knows every other process for the broadcast group and the round-length used in this algorithm could then grow as a function of worst-case network latency. When a message is broadcasted to all members in its group, the size of a vector time stamp piggybacked on the message is the number of the members. Thus, as PCBCast does not need a $N \times N$ matrix for saving the ordering control information, it results in lower communication overheads than previous causal ordering algorithms.

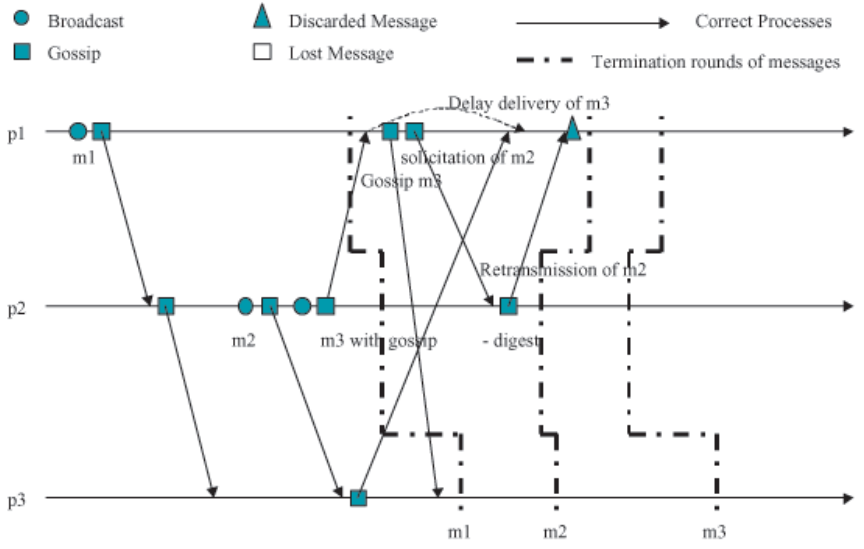


Fig. 1. An *PCBCast* algorithm execution without any failed process or lost message

For example, figure 1 depicts an execution of the algorithm without any failed process or lost message. In our algorithm, sending process p1 generates m1 and sends it to a randomly chosen subset of other nodes, {p2}. When p2 initially receives m1, it gossips to p3. Then, p2 generates m2 and gossips it to p3. On receiving m2, p3 gossips it to p1. When process p2 generates m3 and sends it with a gossip-digest information to p1, p1 can know that it didn't receive m2. Then, p1 solicits the message m2 from its sender p2. But, p1 receives m2 from p3 before receiving it from p2. Thus, p1 discards the latter for guaranteeing the integrity property.

Figure 2 shows an example of a PCBCast algorithm execution with one failed process p3 and one lost message m2. In this figure, process p3 receives m1 and

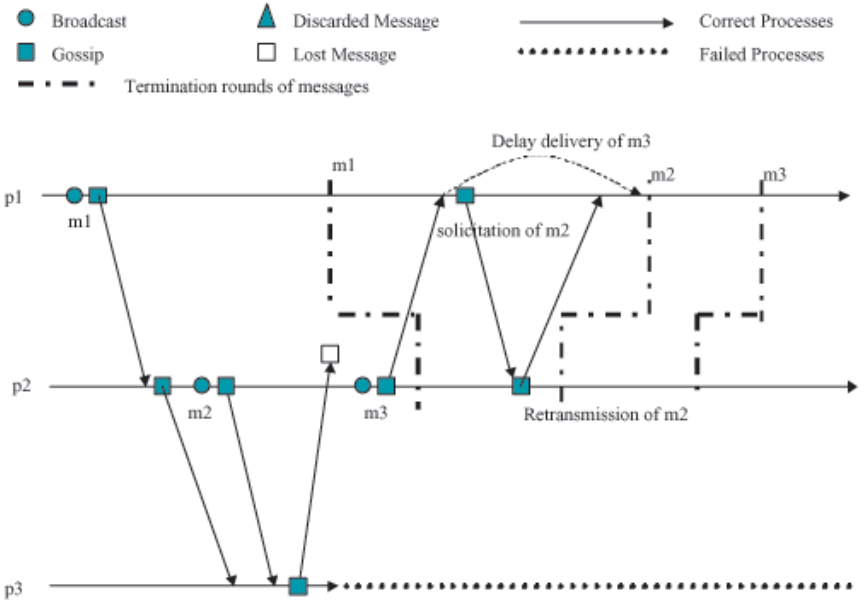


Fig. 2. A *PCBCast* algorithm execution with one failed process p3 and one lost message m2

then m2 from p2 by gossiping respectively. Then, p3 gossips m2 to p1 and fails. Moreover, the sent message m2 is lost. In this case, p1 cannot receive m2 from any other processes. When p2 gossips m3 with digest information to p1, p1 can know that there is the message m2 that causally precedes m3, but has not been received yet. Thus, it delays the delivery of m3. Next, p1 solicits retransmission of m2 from its sender p2. On receiving m2 from p2, p1 delivers m2 and then m3.

Therefore, from the two figures, we can see that any correct process that does not fail can eventually receive broadcast messages by redundant gossiping or solicitation to achieve reliability and fault tolerance.

3.2 Algorithm Description

During the execution of a round, processes gossip broadcast messages with their vector time stamps for ensuring causal order delivery. Each process can gossip messages per round, either for the messages it broadcasts in the round or for the messages some other processes broadcast in the round. Process p maintains a buffer of the messages. Each item in the buffer is a tuple ($senttime$, id, msg, VT, $round$), where $senttime$ is a local time when a message msg is gossiped to other processes, id is an identifier of msg generated by the process broadcasting the message, VT is a vector time stamp and $round$ represents how many times msg is gossiped. The vector time stamp assigned to msg counts the number of messages

that causally precede it. When round R starts, p inserts an element(*senttime*, *id*, *msg*, VT, R) in its buffer. The sender p chooses a random subset of processes to which it will send the message. When process p receives a message from q, it delays delivery of the message until ∀k: $VT(msg)$[k] = VT(p)[k] + 1 (if k=q) or ∀k: $VT(msg)$[k] ≤ VT(p)[k] (otherwise). When a message *msg* is delivered, VT(p) is updated as ∀k: VT(p)[k] = max(VT(p)[k], $VT(msg)$[k]).

```
(* Local state: a message buffer, a vector time-stamp, a vector delivered *)
1:  let buffer = { }
2:  let VT = [ ]
3:  let delivered = [ ]

(Task 1)
4: Actions taken when sending a pcbcast message msg
5:  let id = p
6:  let VT[id] = VT[id] + 1
7:  deliver(msg)
8:  delivered[id] = VT[id]
9:  do_gossip(time(), id, msg, VT, R)

(Task 2)
10: Actions taken when receiving a message msg with (timesent, id, vt, round)
11:  if ((timesent, id, msg, vt, round) in buffer or delivered[id] ≥ vt[id]) then  return
12:  buffer = buffer ∪ {(timesent, id, msg, vt, round)}
13:  do_compare(timesent, id, msg, vt, round)
14:  do_gossip(timesent, id, msg, vt, round)
15:  foreach (timesent, id, msg, vt, round) in buffer
16:    do_compare(timesent, id, msg, vt, round)

(Task 3)
17: Actions taken when handling timeouts
18:  foreach (send, id, msg, vt, round) in buffer
19:    if send + R + 1 == time then
20:      do_deliver(send, id, msg, vt, round)

(Task 4)
21: Actions taken when gossiping a pcbcast message msg with (timesent, id, vt, round)
22:  set_timer = timesent + R + 1
23:  if round == 0 then  return
24:  foreach p in processes
25:    with probability rate
26:      send Gossip(timesent, id, msg, vt, round-1) to p
```

Fig. 3. Procedures for each process p in the PCBCast Algorithm (continued)

Delayed messages are kept in the buffer. This buffer is sorted by vector tie of the messages with concurrent messages ordered by the time of their receipt. If the maximum number of rounds has elapsed after they were received, they must be delivered in their receipt order and then garbage-collected. Figures 3 and 4

describe the PCBCast algorithm. Tasks 1 through 6 execute concurrently, but there is only one instance of each task executing at a time. We assume that the task scheduler is fair, i.e., all tasks get equal chances to execute. Moreover, each line is executed atomically.

```
(Task 5)
27: Actions taken when delivering the messages in its buffer to respect causal order
28:   delivered_ready = true
29:   for all i do
30:     if (i == id or VT[i] < vt[i]) then
31:       delivered_ready = false
32:       break
33:   if delivered_ready == true then
34:     do_deliver(send, id, msg, vt, round)

(Task 6)
35: Actions taken when delivering a message msg with (send, id, vt, round)
36:     deliver(msg)
37:     delivered[id] = vt[id]
38:     buffer = buffer - {(send, id, msg, vt, round)}
39:     for all i do
40:       VT[i] = max(VT[i], vt[i])
```

Fig. 4. Procedures for each process p in the PCBCast Algorithm

When process p starts its execution, a vector time stamp for p is initialized to zeros, and a buffer for gossip messages and a vector *delivered* are created (line 1, 2, and 3). Whenever p broadcasts each message, VT[p] is incremented by one (line 6). Each message p broadcasts is time stamped with the value of VT (line 9). Process p gossips *msg* with its identifier and VT (line 21) and resets *senttime* by adding the sending time and (R + 1)(line 22). When p receives a message from q(line 10), it compares the broadcast message with the messages in its own buffer (line 11). If p does not have or deliver the broadcast message yet, p inserts it into the buffer(line 12). In order to deliver the message on receiving it in causal order within their gossip limitation, p checks its vector time stamp and if the condition is satisfied, the message is delivered and removed from the buffer(line 27-34). For the integrity property, the vector *delivered* should be updated to prevent processes from systematically delivering previously delivered messages again (line 37). If the gossip round is the last, the message is garbage-collected (line 23). Otherwise, the process will continue to gossip about the message to randomly $\beta \times |N|$ chosen members until the maximum number of rounds after its initial reception has terminated (line 24-26), where β is the probability that a process gossips to each other process. Then, messages that remain in the buffer for (R + 1) rounds must be delivered and removed from the buffer (line 18-20).

4 Related Work

Gossip based algorithms are a class of epidemiologic algorithms, which have first been developed for replicated database consistency management[6]. More recently, this class of protocols has been used to build failure detection mechanisms[22], garbage collection[12], leader election algorithms[13]. A recent protocol called pbcast[3,14] uses a gossip-based mechanism for reliable broadcast in large networks. In pbcast, notifications are first broadcasted using either IP multicast or a randomly generated multicast tree if IP multicast is not available. In addition, each node periodically chooses a random subset of processes and sends them a digest of the most recent messages. Upon receipt of these messages, receivers check for missing messages and, if needed, solicit retransmission of them.

Directional Gossip[17] is a protocol especially targeted at wide area networks. By taking into account the topology of the networks and the current processes, optimizations are performed. More precisely, a weight is computed for each neighbor node, representing the connectivity of that given node. The larger the weight of a node, the more possibilities exist thus for it to be infected by any node. The protocol applies a simple heuristic, which consists in choosing nodes with higher weights with a smaller probability than nodes with smaller weights. That way, redundant sends are reduced. The algorithm is also based on partial view, in the sense that there is a single gossip server per LAN which acts as a bridge to other LANs. This however leads to a static hierarchy, in which the failure of a gossip server can isolate several processes from the remaining system.

Felber et al.[8] proposes a specification of probabilistic atomic broadcast with probabilistic safety and liveness properties. This protocol tolerates a configurable number of crash failures, and delivers messages reliably and in order with high probability.

In [7], a fully decentralized membership protocol is presented. Nodes periodically gossip a set of subscriptions they heard about during the last period to a random subset of other nodes, chosen from their partial view. A node receiving such a gossip message updates its partial view by replacing a randomly chosen node-id with a newly received one and gossips the node-id removed from its partial view. While this mechanism achieves a good randomization of the partial views, the size of the partial view and the number of gossip targets are fixed a priori, which precludes decentralized adaptation to changes in system size. Our algorithm, PCBCast, can also run on this membership protocol.

5 Conclusion

In this paper, we present a gossip-based causal ordering broadcast algorithm, PCBCast, guaranteeing weaker, but reasonable reliabilities based on peer to peer interaction models. To guarantee causal order delivery, processes propagate each message with a vector time stamp in an epidemic style for a fixed number of rounds. Upon receipt of these messages, correct processes immediately deliver

the messages to the application layers in such a way that these deliveries respect causal order. As the size of the vector time stamp piggybacked on the message is the number of members in its group. PCBCast significantly reduces the amount of the ordering control information and incurs lower communication overheads compared with previous causal ordering algorithms. Thus, this algorithm can preserve the highly concurrent nature of large-scale P2P systems. For future work, we are currently implementing the PCBCast algorithm on a particular peer to peer system based on a realistic network to demonstrate that both high reliability and scalability can be achieved.

References

1. K. P. Birman, T. A. Joseph. Reliable Communication in the Presence of Failures. *ACM Transactions on Computer Systems*, 5(1), pp.47-76, 1987.
2. K. P. Birman, A. Schiper and P. Stephenson. Lightweight causal and atomic group multicast. *ACM Transactions on Computer Systems*, 9(3), pp. 272-314, 1991.
3. K. P. Birman, M. Hayden, O. Ozkasap. Z. Xiao, M. Budiu, Y. Minsky. Bimodal Multicast. *ACM Transactions on Computer Systems*, 17(2), pp.41-88, May 1999.
4. M. Castro, P. Druschel, A-M. Kermarrec, and A. Rowstron. SCRIBE: A Large-Scale and Decentralized Application-Level Multicast Infrastructure. *IEEE Journal of Selected Areas in Communications*, 20(8), Oct. 2002.
5. F. J. N. Cosquer and P. Verissimo. Survey of selected groupware applications and supporting platforms. *Technical Report RT-21-94*, INESC.
6. A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehar, and D. Terry. Epidemic algorithms for replicated database maintenance. *In Proc. of the 6th ACM Symposium on Principles of Distributed Computing*, pp.1-12, VanCouver, BC. Canada, Aug. 1987.
7. P. Eugster, S. Handurukande, R. Guerraoui, A.-M. Kermarrec, and P. Kouznetsov. Lightweight probabilistic broadcast. *In Proc. of the International Conference on Dependable Systems and Networks*, pp.443-452, July 2001.
8. P. Felber, F. Pedone. Probabilistic Atomic Broadcast. *Hewlett-Packard Technical Report HPL-2002-69*, 2002.
9. S. Floyd, V. Jacobson, C-G. Liu, S. McCanne, L. Zhang. A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing. *IEEE/ACM Transactions on Networking*, pp.784-803, Dec. 1997.
10. A.J. Ganesh, A.-M.Kermarrec, L. Massoulie. SCAMP: Peer-to-peer lightweight membership service for large-scale group communication. *In Proc. of the 3d International Workshop on Networked Group Communications*, London, UK., Nov. 2001.
11. A. J. Ganesh, A-M. Kermarrec and L. Massoulie. HiScamp: self-organizing hierarchical membership protocol. *In Proc. of the 10th European ACM SIGOPS WorkShop*, Sept. 2002.
12. K. Guo, M. Hayden, R. V. Renesse, W. Vogels, K. P. Birman. GSGC:An Efficient Gossip-Style Garbage Collection Scheme for Scalable Reliable Multicast. *Technical Report TR97-1656*, Cornell University, Computer Science, Dec. 1997.
13. I. Gupta, R van Renesse, and K. P. Birman. A probabilistically correct leader election protocol for large groups. *In Procs of the 14th International Symposium on Distributed Computing*, p.89-103, Toledo, Spain, Oct. 2000.

14. M. Hayden and K. Birman. Probabilistic broadcast. *Technical Rerport TR96-1606*, Cornell University, Computer Science, Sept. 1996.
15. D. Kempe and J. Kleinberg. Protocols and impossibility for gossip-based communication mechanisms. *In Proc. of IEEE Symposium on Foundations of Computer Science*, pp.417-480, Vancouver, Canada, Nov. 2002.
16. A.-M. Kermarrec, L. Massoulie, and A.J. Ganesh. Probabilistic reliable dissemination in large-scale systems. *IEEE Transactions on Parallel and Distributed Systems*, 14(3), pp.248-258, March 2003.
17. M.-J. Lin and K. Marzullo. Directional gossip: Gossip in a wide-area network. *Technical Report CS1999-0622*, University of California, San Diego, Computer Science and Engineering, June 1999.
18. D. S. Milojicic, V. Kalogeraki, R. Lukose, K. Nagaraja, J. Pruyne, B. Richard, S. Rollins and Z. Xu. Peer-to-Peer Computing. *Technical Report HPL-2002-57*, HP Laboratories, Palo Alto, March 2002.
19. S. Paul, K. Sabnani, J. Lin, and S. Bhattacharyya. Reliable Multicast Transport Protocol(RMTP). *IEEE Journal on Selected Areas in Communications*, 15(3):407-421, Apr. 2000.
20. D. Pendarakis, S. Shi, D. Verma, and M. Waldvogel. ALMI: An application level multicast infrastructure. *In Proc. of the 3rd USNIX Symposium on internet Technologies and Systems*, pp.49-60, San Francisco, CA, USA, March 2001.
21. S. Ratnasamy, M. Handley, R. Karp, and S. Shenker. Application-level multicast using content-addressable networks. *In Proc. of the 3rd International Workshop on Networked Group Communication*, Nov. 2001.
22. R. V. Renesse, Y. Minsky, M. Hayden. A Gossip-Style Failure Detection Service. *Technical Report TR98-1687*, Cornell University, May 1998.

Intermediate View Synthesis from Stereoscopic Videoconference Images

Chaohui Lu, Ping An, and Zhaoyang Zhang

School of Communication & Information Engineering, Shanghai University, Shanghai 200072,China
llvch@hotmail.com, anping@mail.shu.edu.cn, zhyzhang@yc.shu.edu.cn

Abstract. A procedure is described for stereoscopic videoconference system with viewpoint adaptation. The core to such a system is to synthesize the intermediate views from stereoscopic videoconference images with rather large baseline. The foreground object is first segmented by using intensity and disparity information. For this purpose, the region growing technique is used. The reliability of disparity estimation is then measured with a criterion based on uniqueness and smoothness constrains. In occluded areas and image points with unreliable disparity assignments, region-based interpolation strategy is applied to compensate the disparity values. Finally, an object-based interpolation algorithm is developed to synthesize arbitrary intermediate views. Experimental results with natural stereoscopic image pairs show that the proposed method can obtain the intermediate views with high quality.

1 Introduction

Multimedia systems raise the demand for new interactive communication capabilities between remote partners. Telepresence videoconferencing systems give the user an illusion of true contact, bringing participants together in a virtual space. When he moves his head, a user should be able to acquire corresponding views, and thus "see around" the conferees or 3D objects. One main principle of this technique is the synthesis of intermediate views not acquired by physical cameras. At each station, such physical cameras can only be place around the display, which means that the baseline between the cameras is at least 50 cm with a small display and 80 cm with a larger display. The extreme differences between left- and right-view images do not correspond to the small distance of human eyes, and the resulting stereo presentation would perturb the viewer. Hence, it is necessary to synthesize intermediate views with a smaller baseline.

Several methods [1-7] have been proposed for an arbitrary intermediate view synthesis from stereoscopic images. The strategy behind these methods is similar. The pixel correspondences between the left and the right images are first established. Then, by scaling of the disparity field, an arbitrary intermediate view is interpolated

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 243–250, 2004.
© Springer-Verlag Berlin Heidelberg 2004

from the stereoscopic images. The image quality of the synthesized intermediate view depends mainly on the accuracy of the disparity estimates. In fact, due to camera noise, occluded areas, and lack of image texture, stereo matching algorithm generally cannot provide a reliable disparity estimate for every image point. Therefore the first main difference among these methods is how to deal with these unreliable areas. Uniqueness constraint is mostly used to assess the reliability of disparity estimates [1,2]. In addition to uniqueness constraint, stereo-motion consistency and the analysis of the curvature of the correlation are also employed to the reliability measure [3,4]. In method [5], a criterion is proposed based on a-posteriori probability taking displaced image intensity differences and the variation of disparity estimates into account. Method [1] extracts objects and performs a bilinear interpolation for dense disparity field. Method [2] handles occluded areas by making reasonable assumptions of depth constant. Methods [3,4] apply an edge-assisted disparity interpolation method to the occluded areas and image points with unreliable disparity assignments. Method [6] exploits n (larger than 2) different views to cope with the occluded problem. The second main difference is how new intermediate views are synthesized from stereoscopic images. In method [6], an arbitrary intermediate view is linearly interpolated. Methods [3,4,5,7] synthesize the intermediate view using a nonlinear interpolation. Methods [3,4] use different weighting factors according to the position of the pixel to be interpolated. In method [5], an arbitrary intermediate view is reconstructed with an adaptive, object-based disparity-compensated interpolation. Method [7] exploits the so-called Winner-Take-All strategy to reconstruct the intermediate view.

In this paper, a hybrid algorithm using the intensity and disparity information is proposed to segment the foreground object. For this purpose, the region growing technique is employed. In the process of disparity estimation, a block-matching algorithm considering smoothness constraint is used. To measure the reliability of disparity estimates, we propose to use a criterion based on uniqueness and smoothness constrains. Region-based interpolation is applied to handle the occluded areas and image points with unreliable disparity assignments. Inspired by [3,4], arbitrary intermediate views are synthesized with a simple object-based disparity interpolation. Our approach assumes typical videoconferencing situations, where the head and shoulder part of a person in front of a uniform background form the scene to be processed.

This paper is organized as follows. In Section 2, the algorithm for reliable disparity estimation is described. The method for object-based interpolation synthesis is introduced in Section 3. Section 4 provides some experimental results with natural stereoscopic image pair. Conclusions are drawn in Section 5.

2 Disparity Estimation Algorithm

Throughout the paper, a setup where the cameras have parallel optical and vertical axes and coplanar plans is assumed. This means that only a one-dimensional search along the scan lines is necessary for disparity estimation. In addition, several suitable constraints such as uniqueness and smoothness can be applied in disparity estimation in order to enhance efficiency and accuracy.

2.1 Object Segmentation

Object segmentation plays a crucial role in the process of embedding natural video into virtual or natural environment, in synthesizing the intermediate view, and especially in enhancing the disparity estimates. For this reason the segmentation of the videoconference image into foreground and background areas seems reasonable. The later process is then applied to foreground object only. Based on the facts that disparity segmentation generally yields unreliable results on object boundaries with depth jumps, but reliable results within objects, whereas intensity-based segmentation is usually over-segmented within smooth objects with strong textures, but yields exact boundaries on true object boundaries with strong luminance change, we combine both segmentation results. Disparity estimation is accomplished by means of a block matching algorithm [8], in which the cost function is designed based on the search for similarity of corresponding block together with a smoothness term. The region growing technique [9] is applied to both the disparity map and intensity image to segment into individual regions.

2.2 Reliability Measurement of Disparity Estimation

By the method mentioned above, we get both the left to right disparity map (d^{LR}) and right to left disparity map (d^{RL}) of the foreground object. Note that d^{LR} and d^{RL} have different signs. In order to obtain more accuracy disparity estimates, we propose using a criterion based on the uniqueness constraint together with the smoothness constraint to measure the reliability of disparity estimates. According to the uniqueness principle, each image point may be assigned at most one disparity value. Using the relation

$$\delta = | d^{LR}(x,y) + d^{RL}(x + d^{LR}(x,y), y) \ . \tag{1}$$

the uniqueness condition can be test for each sampling position (x,y). The deviation δ can be seen as a measure of the estimate perturbation and it describes the difference given by the relation (1). In order to consider how much input image satisfy the uniqueness constraint, the following reliability function is employed

$$f_1 = \begin{cases} e^{-C\delta}, & if \quad 0 \le \delta \le A \\ 0, & else \end{cases} . \tag{2}$$

where A is an upper bound for the δ and C is a dampening parameter to control the decreasing velocity of f_1.

The smoothness constraint describes that disparity values vary smoothly in neighborhood region. So we calculate the disparity change

$$\Delta(x,y) = | d(x+L,y) - d(x-L,y) | \ . \tag{3}$$

where $d(x,y)$ is the disparity of position (x,y) and L is window length. We define the smoothness reliability

$$f_2 = \begin{cases} \dfrac{T - \Delta(x,y)}{T}, & if \quad 0 \le \Delta(x,y) \le T \\ 0, & else \end{cases} . \tag{4}$$

where T is a threshold. We define final reliability function f as a linear combination of f_1 and f_2

$$f = \lambda_1 f_1 + \lambda_2 f_2 \ . \tag{5}$$

where λ_1 and λ_2 are weigh coefficients satisfying the relation $\lambda_1 + \lambda_2 = 1$. The function values of f are larger if the reliability is higher, whereas f will take small values for lower reliability.

2.3 Region-Based Interpolation

It is assumed that each region in the image has smooth disparity values. Therefore, the above regions acquired by intensity-based segmentation are used again. For those image points with low reliability, region-based interpolation is introduced to recovery these disparity values using the correctly estimated disparity values within the individual region. Let l_k denote the distance between the position of the disparity value d to be interpolated and the position of the known disparity value d_k for $1 \leqslant k \leqslant 4$. The region-based interpolator presented in [10] is employed

$$d = \frac{w_1(l_3 + l_4) + w_2(l_1 + l_2)}{l_1 + l_2 + l_3 + l_4} \ . \tag{6}$$

where $w_1 = \dfrac{d_1 l_2 + d_2 l_1}{l_1 + l_2}$ and $w_2 = \dfrac{d_3 l_4 + d_4 l_3}{l_3 + l_4}$.

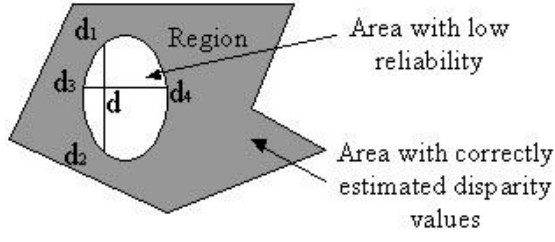


Fig. 1. Region-based interpolation of disparity values

3 Object-Based Intermediate View Synthesis

We want to interpolate intermediate views at arbitrary points along the baseline axis between the left- and right-view images. We define the parameter α=0 as the position of the left, and α=1 as the position of the right image. Hence, any $0<\alpha<1$ is a valid position of an intermediate view. Because of parallel camera setup, the disparity is linear relation with the displacement α of virtual intermediate view. For each position (X_i,Y) on the intermediate image, if the two corresponding positions (X_l,Y) (in the left image) and (X_r,Y) (in the right image) are known, the synthesis of intermediate view can be done by interpolation as follows

$$I_I(X_i,Y) = w_L I_L(X_l,Y) + w_R I_R(X_r,Y) \ . \tag{7}$$

where w_L and w_R are weights of the left and right image.

Due to more visual left part of the object exists in the left image, and vice versa, the right image has more visual right part, d^{LR} can get more reliable result at right part when the left image is referenced, while the d^{RL} shows better image quality at the left part. Hence, we use d^{RL} when point (X_i,Y) is in the left part of intermediate view foreground. When it is in the right part, d^{LR} is used. The equivalent position of the foreground in the intermediate image must be found. This can be done by using the disparities near the border of the foreground in left or right image. The foreground region within the intermediate image is then subdivided into four areas of equal width in turn from left to right, and the two weights are defined according to areas and displacement α.

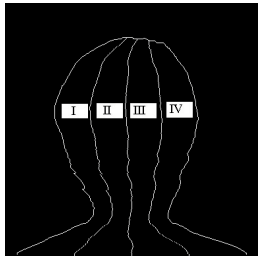


Fig. 2. Foreground contour in intermediate image and four areas with α=0.50

$$\text{area “I”:} \qquad w_L = \begin{cases} 1 & for \quad 0 \le \alpha \le 0.5 \\ 2-2\alpha & for \quad \alpha > 0.5 \end{cases} \ . \tag{8}$$

$$\text{area “II” and “III”:} \qquad w_L = 1-\alpha \ . \tag{9}$$

area "IV": $$w_L = \begin{cases} 1-2\alpha & for \quad 0 \le \alpha \le 0.5 \\ 0 & for \quad \alpha > 0.5 \end{cases} . \tag{10}$$

The weight w_R is set to 1-w_L. Therefore, if point (X_i,Y) is belong to area "I" or "II", the virtual view is synthesized by assigning the intensity value

$$I_I(X_i, Y) = w_R I_R(X_r, Y) + w_L I_L(X_r + d^{RL}(X_r, Y), Y) . \tag{11}$$

where X_i=X_r+(1-α)d^{RL}(X_r,Y). If point (X_i,Y) is belong to area "III" or "IV", the virtual view is synthesized by assigning the intensity value

$$I_I(X_i, Y) = w_L I_L(X_l, Y) + w_R I_R(X_l + d^{LR}(X_l, Y), Y) . \tag{12}$$

where X_i=X_i+α d^{LR} (X_i,Y).

In the process of view synthesis, if the intermediate pixel is mapped by more than one disparity, the disparity with larger value is used, which indicates that positions representing object points close to the cameras are assumed to occlude positions representing object points that are more distant from the cameras. If no disparity values map to a particular intermediate pixel, the missing pixels are obtained using a bilinear interpolator.

4 Experimental Results

The proposed algorithms have been tested with several stereoscopic videoconference image pairs. Here, the results of the third frame pair MAN, which was captured using a parallel camera setup, are illustrated as an example. Fig.3 shows the regions of intensity-based segmentation and the foreground object running on its left image. Each region is represented by different intensity value. It is clear that the object is segmented finely. Fig. 4 shows the result obtained by applying our synthesizing algorithm to the MAN pair. In our experiments, some important parameters for reliability measurement of disparity estimation are shown in Table 1. Here, a constant intensity value that equals to average value of background pixels in the left image is assigned to the background of intermediate views for view convenience. The proposed algorithm provides high quality intermediate views with realistic synthesis, and no visual artifacts are perceivable. This is corroborated by the intermediate views with gradual displacements, which start from one image of the stereo pair and slowly 'warp' towards the other.

Table 1. Parameters for disparity reliability measurement

A	C	L	T	λ_1	λ_2
15	0.1	5	8	0.6	0.4

(a). Regions

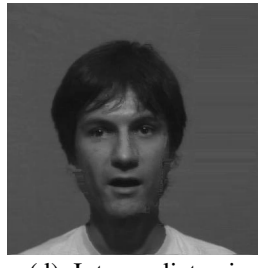
(b). Object

Fig. 3. Regions and foreground object

(a). Original left view

(b). Intermediate view for α =0.25

(c). Intermediate view for α =0.50



(d). Intermediate view for α =0.75

(e). Original right view

Fig. 4. Results of intermediate view synthesis

5 Conclusions

A procedure to synthesize the intermediate views was presented for stereoscopic videoconference system. In the proposed algorithms, a hybrid method was first introduced for the foreground object segmentation. In order to acquire accurate disparity

estimates, the reliability of disparity estimation was measured and region-based interpolation strategy was applied to compensate the disparity values of image points with unreliable disparity assignments. Finally, a simple object-based interpolation algorithm was developed to synthesize arbitrary intermediate views. The performance of the presented methods was tested using natural stereoscopic image pairs. Experimental results show that the proposed method can obtain the intermediate views with high quality, and is capable of offering a realistic 3-D impression in videoconferencing situations.

Acknowledgments. This work was supported by the National Natural Science Foundation of China under Grant 60202015.

References

1. Jens-Rainer, O., Izquierdo, E.: An Object-based System for Stereoscopic Viewpoint Synthesis. IEEE Transactions on Circuits and Systems for Video Technology, Vol.7 (1997) 801–811
2. McVeigh J., Siegel M., Jordan A.: Intermediate View Synthesis Considering Occluded and Ambiguously Referenced Image Regions. Signal Processing: Image Communication, Vol. 9 (1996) 21–28
3. Izquierdo, E.: Stereo Matching for Enhanced Telepresence in Three-Dimensional Videocommunications. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 7 (1997) 629–643
4. Izquierdo, E.: Stereo Image Analysis for Muti-viewpoint Telepresence Applications. Signal Processing: Image Communication, Vol. 11 (1998) 231–254
5. Zhang, L., Wang, D., Vincent, A.: Reliability Measurement of Disparity Estimates for Intermediate View Reconstruction. Proceedings of IEEE Int. Conf. on Image Processing, Vol. 3 (2002) 24–28
6. Werner, T., Hersch, R.D., Hlavac, V.: Rendering Real World Objects Using View Interpolation. Proceedings of IEEE Int. Conf. Computer Vision, Boston, MA, (1995) 957–962
7. Mansouri, A. R., Konrad, J.: Bayesian Winner-take-all Reconstruction of Intermediate Views from Stereoscopic Images. IEEE Trans. on Image Processing, Vol. 9 (2000) 1–13
8. Nikolaos, D.D., Anastasios, D.D., Yannis, S.A., Klimis, S.N., Stefanos, D.K.: Efficient Summarization of Stereoscopic Video Sequences, IEEE Trans. Circuits Sys. Video Technol., Vol. 10 (2000) 501–517
9. Xia, L. Z.: Digital Image Processing. Nanjing (1999)
10. Wang, D., Lauzon, D.: Hybrid Algorithm for Estimating True Motion Field. Optical Engineering, Vol. 39 (2000) 2876–2881

Extract Shape from Clipart Image Using Modified Chain Code – Rectangle Representation

Chang-Gyu Choi[1], Yongseok Chang[1], Jung-Hyun Cho[2], and Sung-Ho Kim[3]

[1] Dept. of Computer Engineering Kyungpook National University, Daegu, Korea
{cgchoi, ysjang}@borami.knu.ac.kr
[2] Division of Computer Technology, Yeungnam Collage of Science & Technology, Daegu, Korea
petercho@ync.ac.kr
[3] Dept. of Computer Engineering Kyungpook National University, Daegu, Korea
shkim@knu.ac.kr

Abstract. This paper presents a method of extracting shape information from clipart images, then measuring the similarity between different clipart images using the extracted shape information. To represent the shape of clipart images, the proposed method expresses the convex and concave aspects of an outline using the ratio of a rectangle. The shape outline us then expressed using a rectangle representation and converted into a chain code. Experimental results demonstrated that the proposed method is superior in expressing shape information than previous outline-based feature methods.

1 Introduction

With the increasing use of the Internet and rapid development of multimedia technologies, for example, the MPEG standard and Wireless technology etc, there is also a need for technologies that can automatically extract, store, transmit, and search for multimedia data, such as images, moving pictures, or audio data. As such, image content-based retrieval systems have already been developed that can automatically extract, store, and search for images using color, texture, and shape.

The extraction methods that use shape features can basically be categorized as either outline-based methods or region-based methods [1~5]. As regards outline-based methods: Freeman and Davis were the first to represent the outline of an arbitrary shape using a Chain Code in a clockwise or counterclockwise direction [6]. However, since a Chain Code will vary according to rotation and size, several modified methods have also been introduced, including the Derivative Chain Code proposed by Bribeeesca and Guzman [7] and Pace Code proposed by Wu [8]. Subsequently, Persoon and Fu proposed a Fourier transform [9] and the EGFD (Enhanced Generic Fourier Descriptor) was recently introduced in [10]. Rauber and Steiger-Garcao proposed the UNL Fourier Features method [11], whereby a UNL Fourier transform is used to change the Descartes coordinates of a shape into polar coordinates. UNL Fourier features are invariant to size and translation, but variant to rotation. Thus, a 2-D Fou-

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 251–260, 2004.
© Springer-Verlag Berlin Heidelberg 2004

rier transform is used to solve this problem. As a result, these methods are invariant to size, translation, and rotation, although the time cost remains the main shortcoming.

Accordingly, to deal with the weaknesses of the previous methods, the current paper proposes a new method that expresses the convex and concave aspects of an outline using the ratio of a rectangle, thereby expressing superior shape information than previous outline-based feature methods.

The remainder of this paper is organized as follows. Section 2 outlines the proposed system and preprocessing methods. Section 3 describes the shape extraction method, rectangle of representation, chain code, and estimation of similarity. Section 4 presents the experiment results, and section 5 gives some final conclusions and suggests areas for future work.

2 Proposed System and Preprocessing Steps

2.1 Outline of Proposed System

The proposed system extracts shape information from a clipart image, saves a database (offline process), then measures the similarity between clipart images (online process). Fig. 1 shows the structure of the proposed system. The offline process consists of two preprocessing steps: extraction of clipart image outline and its conversion into a polygon. The proposed rectangle representation uses the polygon information, which is then changed into a chain code. Finally, the online process measures the similarity between the features of the query clipart image extracted in the same way as the offline process and those in a database.

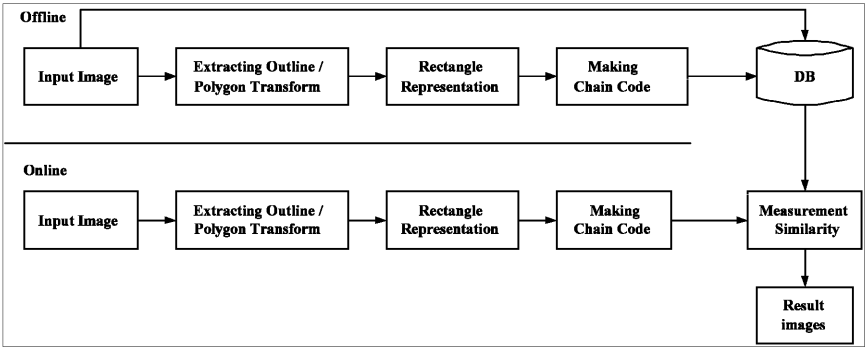


Fig. 1. Diagram of proposed system

2.2 Extraction of Outline and Conversion into Polygon

To extract the outline, a start point pixel is selected in the top-left part of the original image, then sampling pixels are extracted in proportion to the ratio of the image size to make a polygon. The sampling interval is determined by Eq. (1).

$$S = \frac{2(W+H)}{\tau} \tag{1}$$

where S is the sampling interval, W is the image width, H is the image height, and τ is the variable.

Unlike human eyes, an image outline includes noise, which can be removed through the sampling interval. However, there is a trade-off, as if the sampling interval is too large, information will be lost, conversely, if the sampling interval is too small, the noise will be not be removed. Therefore, S is decided by τ and the image size (in the current study, the performance was superior when τ was 45). And, since S is decided according to the size of the image, the proposed method is invariant to size. Starting with the top-left pixel, the pixels located at the sampling interval represent the outline of the image following a clockwise direction.

3 Extraction of Shape Information through Rectangle Representation and Measurement of Similarity

3.1 Rectangle Representation

After extracting the outline and converting the original image into a polygon image, as mentioned in section 2.2, the proposed method uses a rectangle representation to express the convex or concave sections of the polygon image. Figs. 2 (a) and (c) show part of the polygon image, while (b) and (d) show the respective rectangle representation of the concave regions. In Figs. 2 (a) and (c), the concave regions are similar, but (a) is more concave than (c). This difference between the concave regions is clearly represented when using the rectangle representation, even though the shapes are both concave regions. The same method is also adopted for convex regions.

Before proceeding to the rectangle representation, a step is included that decides whether a polygon's vertex is convex or concave in a clockwise direction [12]. The vertex is convex when the next vertex is located to the right in a clockwise direction, otherwise the vertex is concave, i.e., the next vertex is located to the left. Fig. 3 expresses both cases where p_1 is concave and convex, respectively.

To determine p_1, let vectors v_1 and v_2 be those referred to in Fig. 3. To determine whether a vertex is concave or convex, calculate vector v that is determined by the cross vectors, v_1 and v_2. Let points $p_1 = (x_1, y_1, 0)$, $p_2 = (x_2, y_2, 0)$, and $p_3 = (x_3, y_3, 0)$. Eq. (2) shows the vectors v, v_1, and v_2. If v's z value is larger than 0, i.e. the next vertex is located to the right, p_1 is judged as concave, alternatively, if v's z value is lower than 0, p_1 is convex. If p_1 is equal to 0, i.e. point p_1 is located on line, p_1 is determined by p_2's attribute.

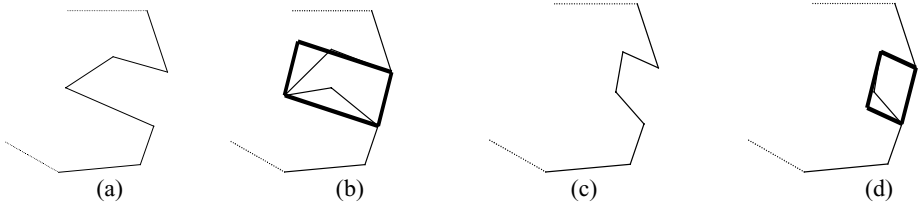


Fig. 2. Polygon and rectangle representation: (a) and (c) polygon images, (b) and (d) rectangle representations, respectively

$$
\begin{aligned}
v &= v_1 \times v_2 = (0,0, x_{v_1} y_{v_2} - x_{v_2} y_{v_1}) \\
v_1 &= (x_{v_1}, y_{v_1}, z_{v_1}) = p_1 - p_2 = (x_1 - x_2, y_1 - y_2, 0) \\
v_2 &= (x_{v_2}, y_{v_2}, z_{v_2}) = p_3 - p_1 = (x_3 - x_1, y_3 - y_1, 0)
\end{aligned}
\tag{2}
$$

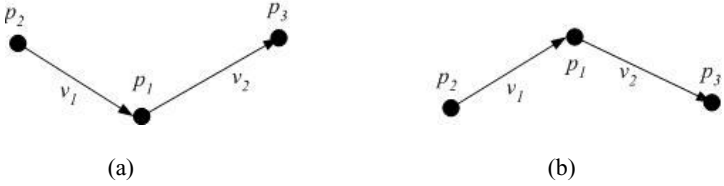


Fig. 3. Relationship among three polygon points: (a) p_1 is concave point and (b) p_1 is convex point

Fig. 4 shows a flow chart of how the rectangles are created from the shape information. Before initiating the steps in Fig. 4, a determination is made whether the polygon's vertexes are convex or concave in a clockwise direction, refer to Eq (2). If the first selected point, starting from the top-left point of the polygon image, is concave, the next search is for the first convex point in a counter clockwise direction. From this convex point to the next convex point, a point by element set is then included one by one in a clockwise direction. If the first selected point is convex, the width and height of the rectangle is calculated, where the width of the rectangle, W, is the length of the end-to-end convex points and the height, H, is the maximum length between the end-to-end convex points and other concave points. These steps are executed until all points are included in the rectangles.

However, the proposed method may have exception handling, i.e. no concave points. Therefore, to solve this problem, all the distances between the points are calculated and the maximum distance, W selected. Based on W, the polygon is segmented into region-A and region-B. The height, H_1 and H_2, is determined based on the maximum length between W and the points included in each region. Finally, height H is added to H_1 and H_2, i.e. $H = H_1 + H_2$. Fig. 5 shows an example.

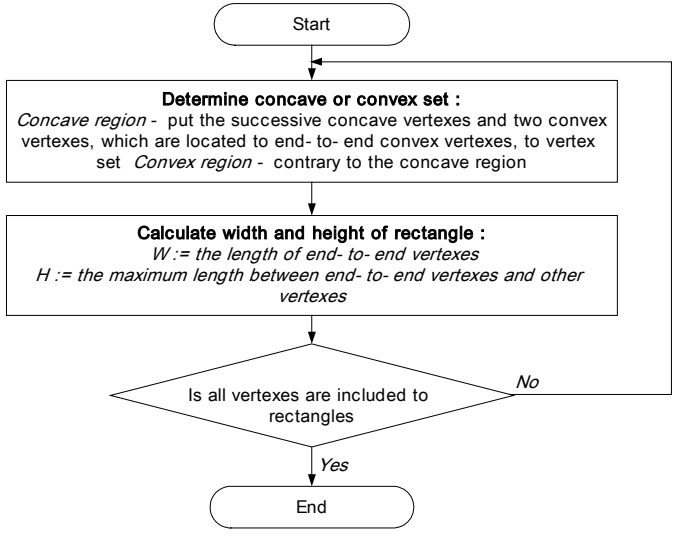


Fig. 4. Flowchart for creating rectangle representation

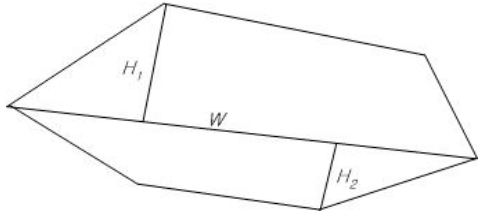


Fig. 5. Results when polygon is convex hull

3.2 Construction of Chain Code

To measure the similarity between the shapes of clipart images, a modified Chain Code is constructed from the rectangle representation using two symbols, one representing a convex rectangle and the other representing a concave rectangle, plus a real value that represents the ratio of the width to the height of the rectangle. The real value is used to express the extent of the convexity or concavity, as shown in Figs. 2 (b) and (d). Symbol *C* is used to represent a concave rectangle, while *V* represents a convex rectangle. The real value is calculated based on the ratio of the height of the rectangle, *H*, to the width of the rectangle, *W*, i.e. *H*/*W*. The width and height are calculated as presented in Fig. 4.

Fig. 6 shows an example of a rectangle representation and the associated chain code. The chain code is constructed from the start rectangle following a clockwise direction and consists of the real value a_i, which is the ratio value, and the symbol *C* or *V* related to each rectangle. As such, the chain code is constructed based on an

arrangement of real values and symbols. For example, the chain code '1V' in Fig. 6 (b) means that the ratio of the rectangle is 1 and that the rectangle is convex.

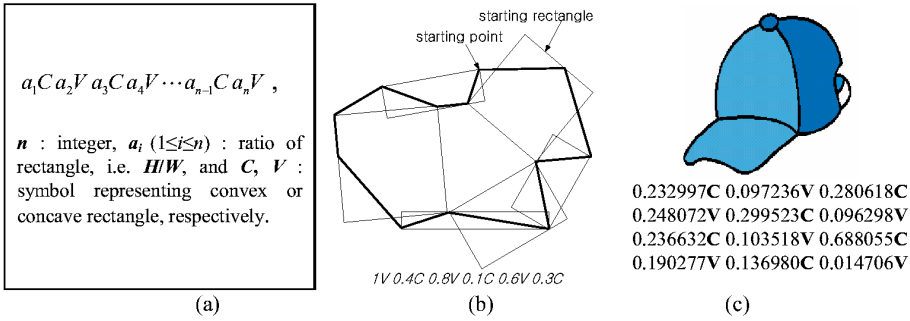


Fig. 6. Example of chain code: (a) form of chain code string, (b) example of converting rectangle representation into chain code and (c) example of chain code

The use of a rectangle representation and chain code produces information that is invariant to size and translation. For example, when two shapes are the same, the same chain codes will be constructed even if their size or location is different. However, the information is not invariant to rotation, as the start point will differ for the outline extraction and translation into a polygon.

3.3 Measurement of Similarity

Using the vectors in the chain code, the similarity step is as follows. First, the same chain code length is created for comparative clipart images, since the length of two clipart images may be different or the same. Second, the similarity between a query clipart image and the database is measured.

To make the same chain code length for two images, the length of the longer chain code between two clipart images is shortened, thereby modifying its shape. In the current paper, to make the same length, concave regions are eliminated, as shown in Fig. 7. However, when a concave region is removed, as in Fig. 7 (b), this results in a loss of shape informat-ion, therefore, a weight is needed when comparing with other clipart images. Let the chain code be represented by Eq. (3). When a concave region is removed, as Fig. 7, Eq. (3) is changed. Then, the concave region is merged with a convex region. To maintain the properties of the polygon, C (or V) and V (or C) must be located successively, $a_{k-1} V a_k C a_{k+1} V$ is merged to dV, and Eq. (4) depicts the revised chain code.

$$S : \cdots a_{k-1} V a_k C a_{k+1} V \cdots, \quad 1 < k < n \tag{3}$$

$$S' : \cdots dV \cdots, \quad d : real\ value \tag{4}$$

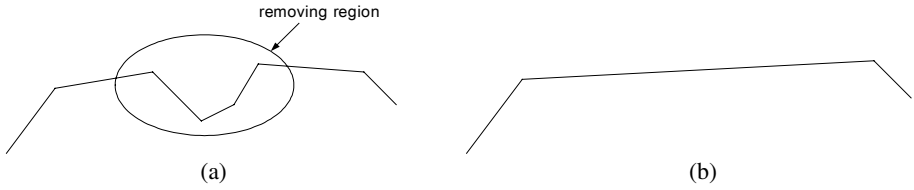


Fig. 7. Example of removing region: (a) state before removing region and (b) state after removing region

The weight must be the minimum value to change Eq. (3) to Eq. (4). To obtain the minimum weight, d is determined as the minimum sum of the distance between d and each point, a_{k-1}, a_k, and a_{k+1}. The minimum weight is thus the minimum cost to modify a concave region, which is located in the center, and the convex regions, which are located at the end-to-end sides. And, a_{k-1}, a_k, and a_{k+1} are regarded as a positive value (a_{k-1}, a_{k+1}) in the case of a convex region, otherwise a negative value (a_k), and the chain code length can be shortened to calculate the minimum weight between d and the three values.

Let A be the weight when changing S in Eq. (3) to S' in Eq. (4) and w_t be the similarity, while d, A, and w_t are defined as follows. This procedure is repeated until the two chain codes are the same length

$$
\begin{aligned}
d &= \begin{cases} \dfrac{a_{k-1} - a_k + a_{k+1}}{3}, & if\ \dfrac{a_{k-1} - a_k + a_{k+1}}{3} \geq 0 \\ 0, & otherwise \end{cases} \\
A &= \sqrt{(d - a_{k-1})^2 + (d + a_k)^2 + (d - a_{k+1})^2} \\
w_t &= \frac{1}{1 + A}, \quad a_{k-1}, a_k, a_{k+1} \geq 0
\end{aligned}
\tag{5}
$$

As the next step, the similarity between the two chain codes with the same length is measured by rotating the two real values for each chain code, i.e. the ratio of the rectangle, and calculating the distance between the two chain codes. Here, the real values of the chain code are considered as a vector, so the distance is calculated between two vectors [1]. As such, the rotation of the chain code is solved by the use of a chain code, and the use of two characters matches the same region, i.e. a convex region or concave region.

Let one chain code be S_1, the other chain code that is rotated to ith be S_2^i, and the length of the chain code be n. The similarity is then defined using Eq. (6), where W_s is a value between 0 and 1.

$$
\begin{aligned}
&S_1 = a_1 C a_2 V \cdots a_{n-1} C a_n V, \\
&S_2^i = b_{2i-1} C b_{2i} V \cdots b_{(2i+n-3) \bmod n} C b_{(2i+n-2) \bmod n} V \\
&v_1 = (a_1, a_2, \cdots, a_n), \quad v_2^i = (b_{2i-1}, b_{2i}, \cdots, b_{(2i+n-2) \bmod n})
\end{aligned}
\tag{6}
$$

$$d_i = \sqrt{(a_1 - b_{2i-1})^2 + (a_2 - b_{2i})^2 + \cdots + (a_n - b_{(2i+n-2) \bmod n})^2}$$

$$w_s^i = \frac{1}{1+d_i},\ W_s = \min\{w_s^i\}, \quad i = 1,2,\cdots,\frac{n}{2}$$

The final similarity between a query clipart image and the database is the sum of W_s and W_t. For normalization, this sum is divided by 2, thus Eq. (7) is the similarity.

$$W = \frac{W_s + W_t}{2} \tag{7}$$

4 Experimental Results

The experiment involved 880 clipart images, consisting of 172 original images, along with the same 172 images reduced 25% and 50%, and rotated 90° and 315°. To compare with other methods, a Chain Code, Fourier descriptor and UNL Fourier descriptor were selected and the performance results summarized in terms of the *Precision* and *Recall* [1]. The query image was selected from the 172 original images and the performance calculated based on the average *Precision* and *Recall* value. The 5 images with the closest similarity were selected.

Fig. 8 shows the system performance based on τ. The shape information was extracted by applying a rectangle representation and the similarity calculated by changing τ. The perfor-mance was superior when τ was 45, as seen in Fig. 8, as too much shape information was lost when τ was too small, and no noise was removed when τ was too large.

To compare with other methods, a modified Chain Code was selected, as proposed by Bribiesca and Guzman, a Fourier descriptor, and UNL Fourier feature, the vector for which uses 100 coefficients to transform a Fourier. Fig. 9 and Table 1 showed the performances of each method, and the proposed method was found to be superior to the other outline-based methods and more insensitive to a variation in the outline.

5 Conclusion

The current paper presented and implemented a system that can extract shape information from a clipart image, save the information in a database (offline process), and measure the similarity between clipart images (online process). The proposed system consists of three steps, creating a polygon image from the original image, rectangle representation of the polygon image, and changing the chain code. The rectangle representation provides a more detailed representation of the shape information than other outline-methods. And, the proposed method is invariant to rotation based on the use of a chain code. The proposed method can be used the recent technologies such as Mobile, MPEG standard, Internet etc.

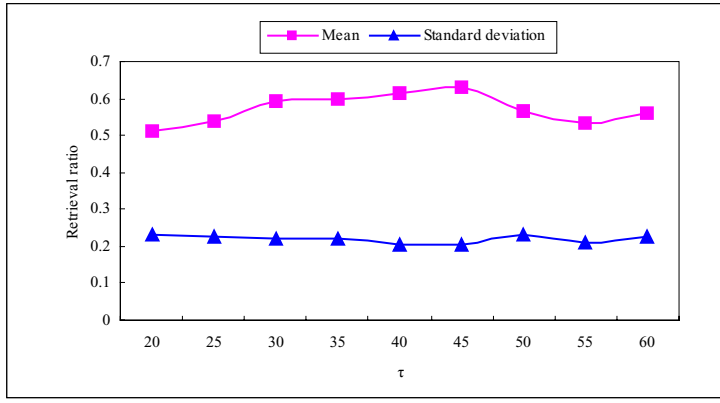


Fig. 8. Standard deviation of retrieval ratio vs. τ

Table 1. Mean retrieval ratio of each method using shape information

Method	Mean retrieval ratio
Proposed method (τ = 45)	0.632
Chain Code	0.473
Fourier descriptor	0.493
UNL Fourier feature	0.509

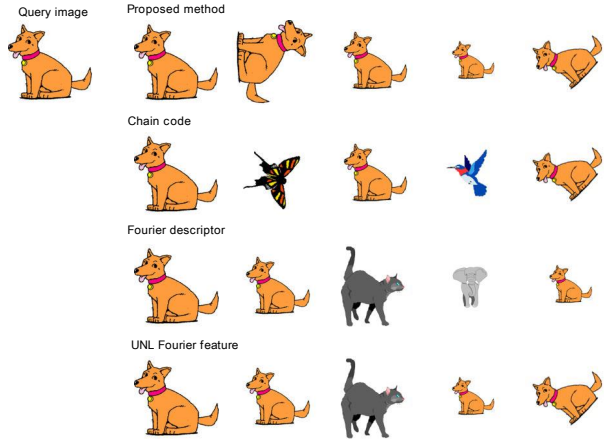


Fig. 9. Experimental result

However, it is difficult to apply the proposed system to natural images, due to problems extracting the outlines, therefore, the new system is presently limited to clipart images. Accordingly, further studies will focus on other features, such as color, texture and the problem of the starting point.

References

[1] B. Mehtre, M. Kankanhalli, and W. Lee, "Shape measures for content based image retrieval: A comparison," Information Processing & Management, vol.33, no.3, pp. 319-337, 1997.

[2] A. Jain and A. Vailaya, "Shape-Based Retrieval: A case study with trademark image databases," Pattern Recognition, vol.31, no.9, pp. 1369-1390, 1998.

[3] Z. Wang, Z.Chi and D. Feng, "Shape based leaf image retrieval," IEE Proc.-Vis. Image Signal Process, vol. 150, No. 1, pp. 34-43, 2003.

[4] G. M. Petrakis and E. Milios, "Matching and Retrieval of distorted an occluded shape using dynamic programming," IEEE Trans. On PAMI, vol. 24, no. 11, pp. 1501-1516, 2002.

[5] A. Folkers and H. Samet, "Content-based image retrieval using Fourier descriptors on a logo database," International Conference on Pattern Recognition, vol. 3, pp. 521-524, 2002.

[6] H. Freeman and L. Davis, "A corner finding algorithm for chain coded curvers," IEEE Trans. on Computers, vol. 26, pp. 297-303, 1977.

[7] E. Bribiesca and A. Guzman, "Shape description and shape similarity for two dimensional region," International conference on Pattern Recognition, 1978.

[8] H. T. Wu, Y. Chang, and C. H. Hsieh, "Coding of arbitrarily shaped objects using pace code," International workshop on Image and Signal Processing and Analysis, pp. 119-124, 2000.

[9] E. Persoon and K. Fu, "Shape discrimination using Fourier descriptors," IEEE Trans. On System, Man and Cybernetics, pp. 170-179, 1977.

[10] D. Zhang and G. Lu, "Enhanced generic fourier descriptors for object-based image retrieval," International Conference on Acoustics, Speech, and Signal Processing, vol. 4, pp. 3668-3671, 2002.

[11] T. Rauber and S. Steiger-Garcao, "Shape description by UNL Fourier features-an application to handwritten character recognition," International Conference on Pattern Recognition, 1992.

[12] F. Preparata and M. Shamos, Computational Geometry, Springer-Verlag New York Inc., 1985.

Control Messaging Channel for Distributed Computer Systems

Bogusław Cyganek and Jan Borgosz

AGH - University of Science and Technology
Department of Electronics
Al. Mickiewicza 30, 30-059 Kraków, Poland
{cyganek,borgosz}@uci.agh.edu.pl

Abstract. The paper addresses the problem of communication among modules of the distributed computer systems. Such systems can be characterized as a connection of many self-contained sub-modules which are either governed by a distinguished unit or are connected in a peer-to-peer fashion. At the same time it is required that the modules communicate reliably with each other without excessive delays and transmission errors. However, very often, due to the nature of embedded systems, these requirements are not matched by the computational power of the system participants. In this paper we present a novel protocol especially designed and implemented for embedded systems. The main advantage of the presented solution consists in the properly balanced proportions between transmission reliability and algorithm complexity. The latter allows implementations starting from the simplest, even 8-bit, microprocessor platforms. There is also no constraint on the used lowest-level protocol layers, electrically or optically connecting components of a system. The paper provides also details of a practical realization in a form of class hierarchies and code.

1 Introduction

In this paper we present a detailed description of the novel communication protocol – the Control Messaging Channel (CMC) – especially designed for computer embedded systems. It is a balanced solution that provides a sufficiently reliable communication for control over components of an embedded system and, at the other hand, that allows for a simple implementation which can fit in even the simplest 8-bit microprocessor sub-systems.

Due to the specifics of embedded systems [1], assuring reliable communication in system is not a trivial task. Especially if the communication means are to be designed to fit different hardware configurations. Starting from the lowest, hardware level we encounter quite different microprocessor architectures (e.g. 8, 16, 32-bit buses, RISC vs CISC, von Neuman's vs. Harvard architectures, etc.), different endian schemes (e.g. PowerPC vs Intel), and different I/O interfaces (e.g. PCI, USB, RS232, Centronics, etc). At the other hand, there are different micro/macro kernels or operating systems running on each of the participating sub-systems [2][3]. Each of those software systems needs to be controlled and needs to communicate with another

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 261–270, 2004.
© Springer-Verlag Berlin Heidelberg 2004

participants of the whole. In such a case it is almost not possible to fit any existing communication standard. Thus designers of the similar systems have to provide custom solutions.

The presented here CMC protocol was developed and implemented to fulfill the aforementioned communication tasks. Its detailed description as well as implementation details are presented in this paper. The CMC has been working in the practical embedded system without any stalls for about one year and showed its great flexibility and reliability.

2 Control Messaging Channel

The main assumptions of the CMC are as follow:

1. CMC connects different modules of the embedded system.
2. CMC is an open architecture; This means that new platforms can be easily connected in the future to the other, CMC aware, components.
3. CMC is a messaging system what means that communication is supported exclusively by message traffic.
4. Messages in CMC are rather control oriented what renders CMC fast and robust. Messages in CMC can transfer data as well (similar to the ISDN control channel [4]). However, such an assumption requires other communication channels e.g. in a case of huge data transfers.
5. Other data transfer paths can co-exist with CMC or even be governed by CMC (similar to the ISDN data channels [4]).
6. Physical layer of the CMC can be any physical connection such as RS-232 [4], USB [5], dual-ported RAM, etc.

2.1 CMC Protocol Description

The CMC protocol relies on the short messages transferred over fixed channels. Structure of the CMC message depicts Fig. 1.

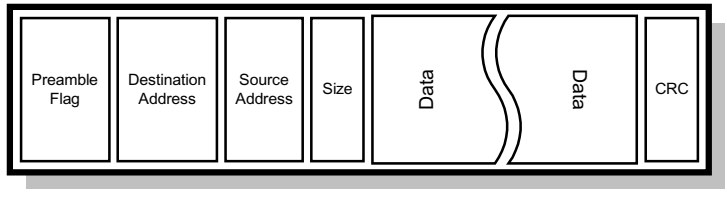


Fig. 1. CMC message structure

Roles of the specific fields of a CMC message are as follows:

1. Preamble Flag:
   Preamble flag contains fixed pattern of bits (0x7E).

0	1	2	3	4	5	6	7
0	1	1	1	1	1	1	0

2. Destination Address:

0	1	2	3	4	5	6	7
System Address			Local Address				

3. Source Address:

0	1	2	3	4	5	6	7
System Address			Local Address				

4. Size:

0	1	2	3	4	5	6	7
Size in **bytes** of the data field							

5. Data:

   Sequence of **0** to **255** bytes.

6. CRC:

0	1	2	3	4	5	6	7
CRC value taken from all fields **except** the preamble field and the CRC field							

The CRC field is computed according to the following polynomial:
CRC-8: $\mathbf{x^8+x^2+x+1}$

Each message conveys the CRC field (by default it is 8 bit long). This field simplifies message authentication as well as simple security against transfer errors. Optionally, especially for time critical transfer actions, the CRC field can be dropped from the CMC message definition.

The CMC protocol assumes that each message contains two address fields:

1. Source address.
2. Destination address.

Each address field itself is divided into two address partitions:

1. System address.
2. Local address.

The first address partition uniquely identifies an CMC system. The second one uniquely identifies a client within that system.

Messages can be of different size (by default from 8*5 bits up to 8*(5+255) bits).
The *size* field contains number of *bytes in the data part* of a message.
The *data* field contains user data (from 0 up to 255 bytes).
Contents of the data field of a message is governed by a user defined higher protocol over CMC; e.g. it can be assumed that the first part of a data field contains message ID, whereas the rest conveys other data for that particular message.

Each CMC message has a *preamble flag* (i.e. header) field used for proper message alignment.

2.2 Bits Stuffing

For proper frame alignment and detection it is necessary to exclude from message bytes those that are identical to the preamble flag bits. Thus, a bit stuffing mechanism is required before a message is put into a transmission channel and then a reverse stuffing process when extracting from this channel. In our implementation a classical preamble doubling algorithms was used [4].

2.3 Higher Protocols

CMC has been designed to facilitate higher protocols performed by means of the CMC protocol. No means has been provided to support advanced protocols mechanisms such as dynamic routing or protocol management, error recovery, etc.

Higher layers of protocols can be designed by specification of the further structure imposed on the *data* field of a CMC message.

3 CMC Architecture and Implementation

For the sake of the multiplatform operation and software reusability, the CMC implementation will be partitioned into two distinct software layers:

1. Platform independent sources – contain classes that do not change across different platforms.
2. Platform specific sources – contain platform specific classes.

For class hierarchies we used the notation introduced by Taligent [6].

Basically there are three sorts of objects involved in CMC mechanism:

1. Clients
2. Dispatchers
3. External Devices

The clients and dispatchers are the CMC building blocks. The overview of the proposed CMC architecture presents Fig. 2. A low-level physical connection is established exclusively among sub-systems' dispatchers and can be of any available type such as DMA, DRAM, USB, RS232, PCI, etc. Optionally, some external devices that are unable to form their own CMC sub-system are governed by specialized CMC clients (adapters).

The message structure from Fig. 1 can be represented by a class in Fig. 3.

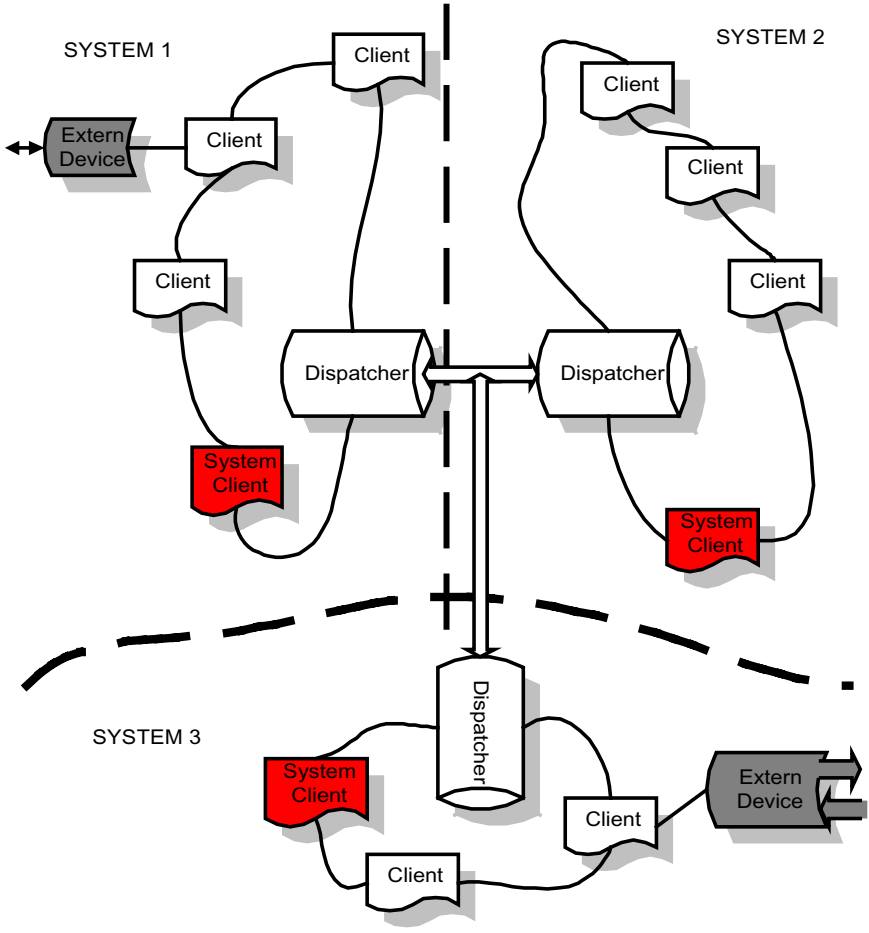


Fig. 2. The CMC implementation architecture. Visible are exemplary three participating systems (divided by a broken line). A low-level physical connection is established exclusively among sub-systems' dispatchers and can be of any available type. External devices that are unable to form their own CMC sub-system are governed by specialized CMC clients

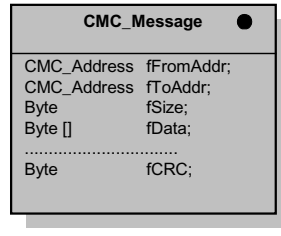


Fig. 3. The CMC message class

3.1 CMC Clients

Clients are objects-interfaces that connect CMC messaging with other system layers. Main functionality of a client is to send messages to other clients and get messages addressed to it.

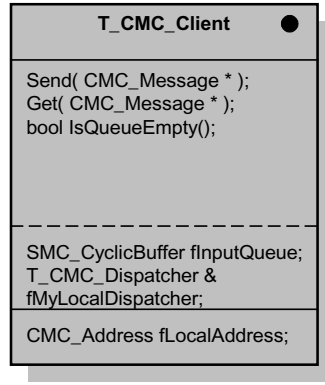


Fig. 4. The CMC client class

Each *T_CMC_Client* object contains three essential components:

1. Cyclic buffer for input messages.
2. Reference to the dispatcher object(s) in which this client is registered.
3. Local address.

There is a special group of clients, called system clients. Their purpose is to govern creation/destruction processes of other clients in the system. Their existence is important for proper communication initiation among different systems. Their interface supports builder or the factory pattern [8] capable of creating (destructing) of other members of the CMC communication channel. It is assumed hereafter that each system supporting CMC messaging has predefined system client and its local address is 0. It is up to the user agreed protocol how the system client responds to the "create-client" requests. One of the possible solutions is to register potential clients in a prototype part of the system client and define special message for creation of one of just registered clients. Each registered client can be identified by a string or other ID object.

It is often desirable to associate an external device, such as "old" system, with an CMC client to make that device available to other participants of the CMC messaging channel. The connection between an external device and the CMC client is local to those two objects (see also Fig. 2).

3.2 CMC Dispatchers

Dispatchers are objects exchanging information on behalf of clients. Their main functionality is to route messages among participating clients, as well as among participating systems of clients, i.e. clients enclosed in other (but somehow connected) system (Fig. 2). The CMC dispatchers' class hierarchy depicts Fig. 5.

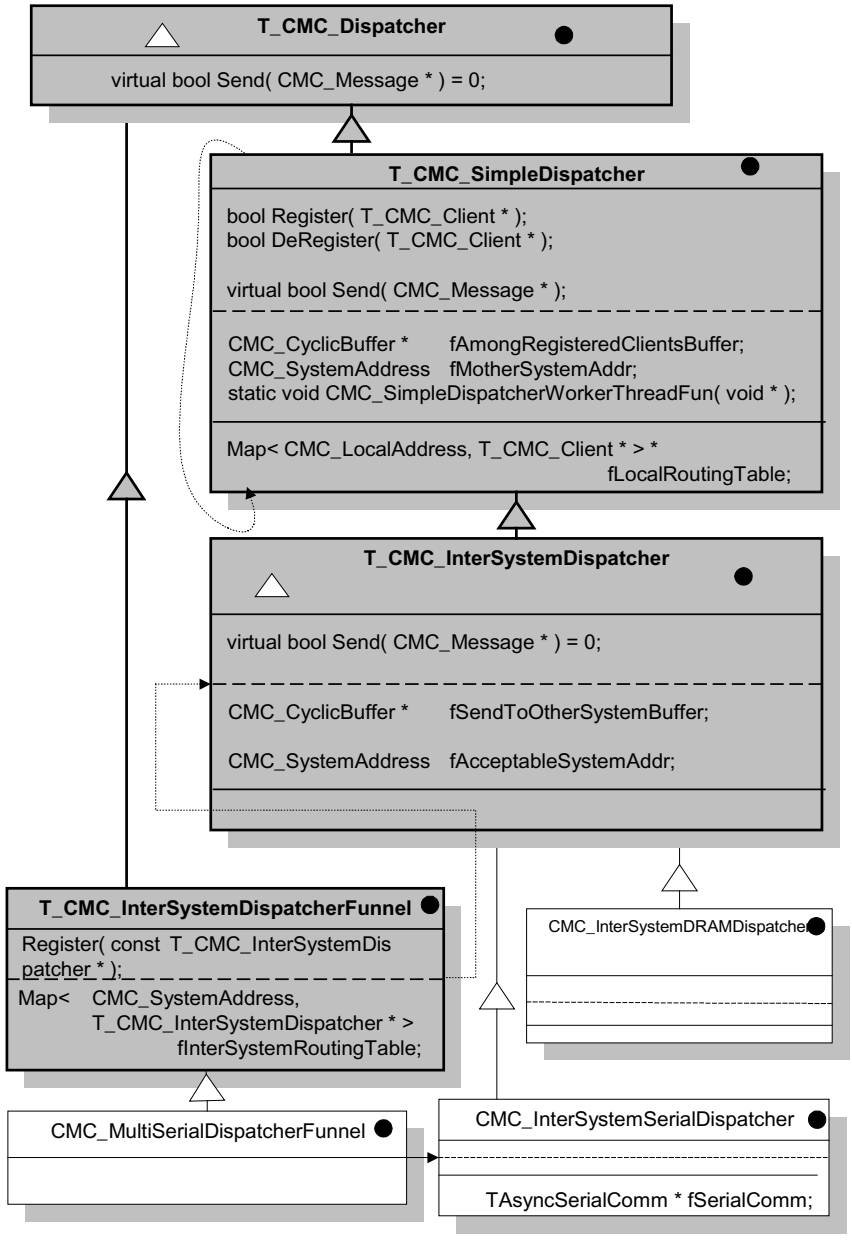


Fig. 5. The CMC dispatchers class hierarchy

There are three types of dispatchers:

1. Simple Dispatchers (1).
2. Inter-System Dispatchers (1:1).
3. Inter-System Dispatcher Funnels (1:n).

The purpose of the first kind is to dispatch messages only in one system, whereas representatives of the second kind can also dispatch messages between two registered systems, and the third one among many registered systems.

Message routing is based on *destination address field* in the CMC message, which in turn is divided into two parts:

1. System
2. Local

The *T_CMC_SimpleDispatcher* interface consists of the following actions:

1. Registering and de-registering clients belonging to the same system, i.e. object of the *T_CMC_Client* class.
2. Sending messages, i.e. *CMC_Message* objects, among registered clients (still belonging to the same system).

The *Send(CMC_Message)* member implementation puts a given message into the *T_CMC_SimpleDispatcher* buffer from where messages are re-distributed to the destination by means of the *T_CMC_SimpleDispatcher* internal mechanisms.

The *T_CMC_SimpleDispatcher* object must perform a systolic action of sending messages currently contained in its cyclic buffer (*fAmongRegisteredClientsBuffer*) to the registered clients in accordance with address field of each message. Messages with wrong address are simply discarded.

The aforementioned systolic action of message distribution can be performed by means of many mechanisms pertained to the operating system such as threads, timers, polling, etc. Care must be taken to ensure thread-safe access to the cyclic queues.

The *T_CMC_InterSystemDispatcher* object is also a kind of the *T_CMC_SimpleDispatcher* object but contains additional queue for message dispatching to the connected system. This system is identified by a system address (*fAcceptableSystemAddress*). The *T_CMC_InterSystemDispatcher* class is pure virtual and as such is a base for specialized dispatchers.

The *T_CMC_InterSystemDispatcherFunnel* object contains one or more registered *T_CMC_InterSystemDispatcher* objects identified by their system addresses. Routing of a message is based on the system address contained in that message. Then the chosen *T_CMC_InterSystemDispatcher* dispatcher object is responsible for further message transfer to the (possibly remote) system.

Another actions are associated with messages reception and re-distribution in the case of many *T_CMC_InterSystemDispatcher* dispatchers working in concordance. In this case, each of the mentioned dispatcher after its internal reception mechanism reports a new message advent (e.g. from a serial link) puts that message in its cyclic buffer (i.e. base member *fAmongRegisteredClientsBuffer*). Then the systolic mechanism (also from the base class) performs proper message distribution among clients of the same system according to its local (local address - client) routing table. The routing table itself should be local. However, it can be copied from one simple dispatcher to the other.

A client object must be at first registered in the dispatcher object to be able to get an access to the CMC mechanisms. After that this client can call *Send(…)* method on its dispatcher. One client can be registered in many dispatchers.

A funnel-dispatcher has been developed to allow for connection of many separate dispatchers into one entity. All clients belonging to these dispatchers are automatically switched to the given funnel rather than their previous dispatchers. This is achieved by means of an internal call to the member:

T_CMC_InterSystemDispatcherFunnel::ChangeClientsAssignment() is called when registering new dispatcher to the funnel. After this *all* clients of the registered dispatchers can send their messages to all connected (via dispatchers) CMC systems. The situation is quite a bit different with reception from the external systems, however. Since messages come to the appropriate dispatchers, only clients registered to this particular dispatcher receive CMC messages from the attached external system. To allow reception from all other systems, i.e. those that are allowed by dispatchers registered into one funnel, all interested clients *should register separately to each dispatcher*. Care should be undertaken to assure that there are unique local numbers among such connected dispatchers.

3.3 CMC Message Queues

A message queue is an auxiliary data structure [1], used by *T_CMC_Client* and *T_CMC_Dispatcher* objects.

The following actions are supported by any *T_CMC_CyclicBuffer* object:

1. *Push back* – inserts an element at the tail of the queue.
2. *Pop front* – gets an element from the front of the queue.
3. *Reset queue* – empties the queue.
4. *Is empty queue* – a predicate, returning true if the queue is empty.
5. *Get queue total size* – returns a total number of bytes reserved for the queue.
6. *Get free space size* – returns number of free bytes in the queue.
7. *Report error* – returns information on errors during queue operation.

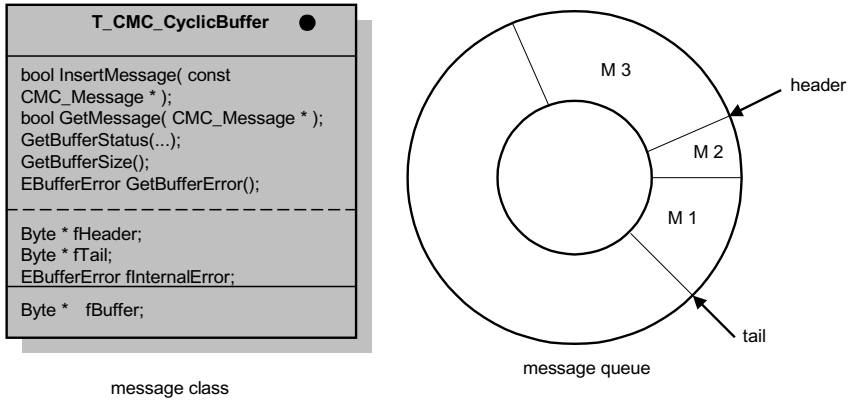


Fig. 6. Structure and class for the CMC queue

Structure of the cyclic buffer presents Fig. 6. There are three main components of the cyclic queue:

1. The support data buffer.
2. The header pointer.
3. The tail pointer.

4 Results and Conclusions

The paper presents a complete Control Messaging Channel for communication among components in embedded computer systems. The detailed description of an implementation was also included.

The presented protocol and implementation was tested in an embedded system connecting three subsystems. The first subsystem consists of the PowerPC 8260 unit governed by a custom micro-kernel. The second subsystem is a StrongARM microprocessor system running Windows CE 3.0. The third subsystem is a simple control panel with the 8052 controller; Its CMC software is written exclusively in assembly. All aforementioned subsystems are connected via RS232 and DPRAM links. The setup was tested for almost a year and showed great reliability and control robustness. It is also planned to connect the first and second subsystems via an Ethernet link to speed-up the communication.

Based on the experimental results and observations we can conclude that the CMC system is very suitable for all moderate size embedded systems that do not require transfers of huge amount of data (for which one can use other transmission channels, such as DMA). Its great usefulness comes from the open architecture – the CMC is able to connect quite different components of an embedded system and its simple implementation can fit almost any microprocessor platform.

References

1. Douglass, B.P.: Doing Hard Time. Developing Real-Time Systems with UML, Objects, Frameworks, and Patterns. Addison-Wesley (1999)
2. Yaghmour, K.: Building Embedded Linux Systems. O'Reilly (2003)
3. Labrosse, J.J: Embedded Systems Building Blocks. R&D Books (2000)
4. Halsal, F.: Data Communications, Computer Networks and Open Systems, Addison-Wesley (1995)
5. USB Org.: Universal Serial Bus Revision 2.0 specification. www.usg.org (2000)
6. Taligent Inc.: Taligent's Guide to Designing Programs: Well-Mannered Object-Oriented Design in C++. Addison-Wesley (1994)
7. Cormen T., et.al: Introduction to Algorithms, Second Edition. MIT Press (2001)
8. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns. Addison-Wesley (1995)

Scene-Based Video Watermarking for Broadcasting Systems

Uk-Chul Choi, Yoon-Hee Choi, Dae-Chul Kim, and Tae-Sun Choi

Department of Mechatronics, Kwangju Institute of Science and Technology,
1 Oryong-dong, Buk-gu, Gwangju 500-712, South Korea
saesang@kjist.ac.kr

Abstract. In this paper, we propose a new scene based video watermarking method by using the DCT. Scene based and DCT based watermarking give many merits in the view of the processing time and quality of the video. Watermarking embed processing is done by following the variance of each DCT components in the scene-changed frame. Low variance components have a stable status. The comparison of video quality following the embedding parts is done and the results show the improvements. MPEG encoder and decoder are simulated in the desktop PC. Results show the proposed algorithm's effectiveness in real time and usage of a buffer and robustness in the watermarking detection.

1 Introduction

Once a de-scrambled video data can be saved in the hard-disk and this video data will be copied without any limitation. To compensate with this limitation, watermarking method is developed. It is conducted by embedding the copy information or noise in the payload. Some of simple method such as adding a logo to the video data could be used, but this has to sacrifice the video quality and could be easily deleted because of visibility. This logo could not cover the all frames, so attacker removes the part of the frames. Many watermarking developer try to make it invisible and hard to erase. In this paper, we will extend these ideas to the broadcasting video that is the most representative. The adaptive watermarking method for the broadcasting video could be considered as blind, real time, robustness and small calculation. These constraints are common to most of other watermarking method.

The most attractive parts of proposed algorithm are that video watermarking is based on the scene change and DCT. Scene is the smallest unit of the video sequence that give a specific and common characteristics. Though shot is smaller than scene, it has a limit regarded as a unit. If we divide sequence to the scene, we easily edit the video and watermarking in a scene has a key value. It reduce the time and space. And DCT based watermarking method is widely used because of adaptive for the Human Visual System (HVS). It is located between the VLC watermarking and raw video data watermarking. The more detail reasons of the advantage would be remarked in the future.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 271–280, 2004.
© Springer-Verlag Berlin Heidelberg 2004

2 The Proposed Method

Before embedding watermarking, we need to classify the scene boundary as a pre-processing. This will save the much time and video quality. Sudden-SCD could be detected by the DCT coefficient [1-4] or motion vector type [5, 6] in the compressed domain. Sudden-SCD by the DCT coefficient is can be detected by the DCT DC value that is mean of luminance value of a frame. There are two representative methods to detect the gradual scene change detection. One is using the DC coefficient's variance [5,7,8] and the other is the difference between the frames that have same interval [9,10]. However, all of methods try to find the ideal case, so they could not cover the all transition whose using frequency is increasing because of the high development of the personal PC and edit tools such as the Premier. Another problem is the method of watermarking embedding processing. Though there are many advantage of VWM based on the DCT, there are some conflict which position is much better to embed the watermarking between AC and DC coefficients [7,8,10]. The best way to embed the watermarking in the compressed domain with scene change is gained by the characteristics of the scene. Sudden or gradual scene change happened at most one second. If the scene changing is detected more than once in a second, human eyes would feel very uncomfortable and even can't detect the scene. So, we detect the scene change as a unit of one or more group of pictures (GOP). After scene changes are detected, we choose the video watermarking (VWM) based on the discrete cosine transform (DCT) that has many merits as the problem of the time and complexity and robustness. Though there are so many kinds of VWM based on DCT, we will propose the simplest way for real time and short storage device. Finally, we will embed the watermarking following the magnitude of the variance that gives information of the noise robustness.

2.1 Scene Change Detection (SCD)

For example, the scene change happens in the frame number 6 as the figure1. In this case the macro block in the 6^{th} frames used the backward prediction. We can define the frame 6 have more similar characteristics with the frame 10 than 1. So we had better to know the frame that have scene change happened.

$$D_i = DC_i - DC_{i-1} \tag{1}$$

This formula shows the first type of the SCD by using the DCT's DC value. Second scene change method is detection of the frame types. Except I frame, there are B and P frames in the GOP. Following the frame type, we have to extract the different ratio of the MB type. For the P frames, there is one ratio between the intra coded macro-block numbers to the forward predicted macro-block numbers.

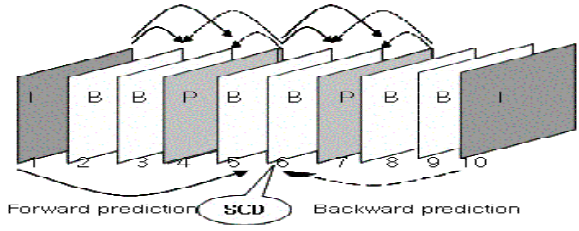
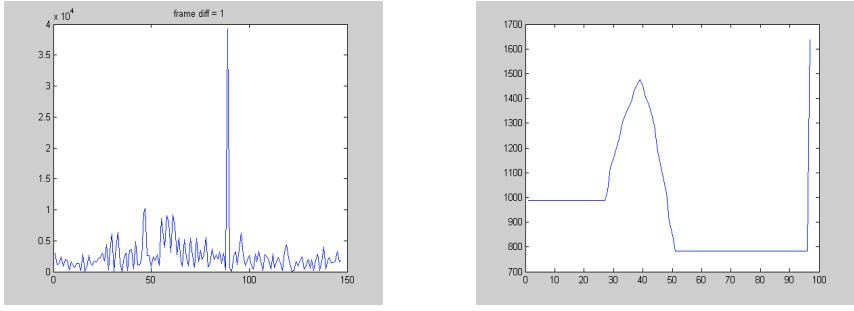


Fig. 1. Hierarchical Composition of the Compressed Video



(a) Sudden SCD (b) Gradual SCD

Fig. 2. DC Difference of Example Video

$$R_{intra1} = \#\text{ of intra coded macro-blocks} / \#\text{ of forward predicted macro-blocks} \quad (2)$$

Different from the P frame, there are three ratios in the B frames. Because B frame can predict forward and backward, there are other macro-blocks that are backward predicted macro-block and interpolated macro-blocks. In the B picture, the number of interpolated MBs with forward and backward motion compensated is proportional to the correlation of previous and next I picture or P picture.

$$R_{interpolated} = \#\text{ of intra coded macro-blocks} / \#\text{ of forward predicted macro-blocks} \quad (3)$$

$$R_{backward} = \#\text{of backward coded macro-blocks}/\#\text{of forward predicted macro-blocks} \quad (4)$$

$$R_{intra} = \#\text{ of intra coded macro-blocks} / \#\text{ of forward predicted macro-blocks} \quad (5)$$

Scene based VWM has an advantage that we can embed the watermarking following the scene's characteristics. Firstly the frames in a scene have almost same luminance and objects. In this view of the common parts, we can guess how to detect the scene boundary in the compressed domain. Secondly the watermarking strength and location is determined by scene complexity as the unit of the scene in a GOP and object's motion magnitude. In this chapter, we will consider the full algorithm with these ideas. Thinking gradual scene change algorithm connecting the sudden-SCD by macro block type, we can find some strange effects. Gradual scene change could be separated by the luminance change, overlap of the reference frame without motion

and with motion. The method in figure 4 can catch the scene change if the gradual scene change is done by the luminance change and the overlap without motion. However, if the gradual scene change using prediction with motion is used for the transition, there are high possibilities that new frame is created without prediction.

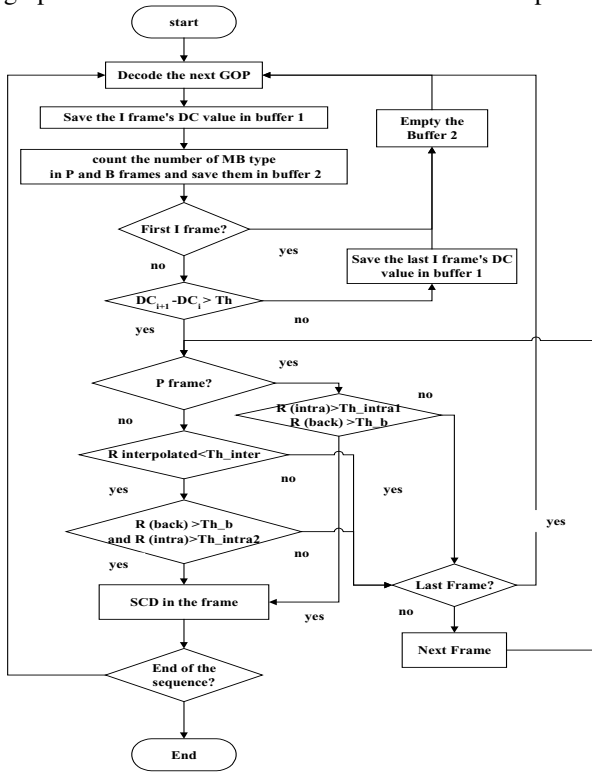


Fig. 3. Flow Chart of the Scene Change Detection

First Step:

Save the picture type and macro block type in a GOP and then subtract successive I frame's DC value and compare with the threshold that is determined already. If the value of subtraction is larger than the threshold, then we suspect a GOP as scene change can be happened. If not, skip to the next I frame.

Second Step:

We can guess that the scene change would be happen just one time in a GOP. So find the best frame that could be regarded as a scene changed frame. As we have explained in the previous work, the simple and fast way of SCD is done by counting the macro block type. And there are two kinds of picture type except I frame. In the case of P picture, we have one parameter of intra block to forward predict block. In the case of B picture, we have three parameters shown in the formula (3), (4), (5).

Third Step:

If the scene changed boundary frame is detected, we skip to the next GOP. And the saved data is removed in the buffer.

Firstly, we have to know a macro block's complexity to know whether the macro block is enough complex to embed the watermark without annoying the HVS. If the macro block's complexity high enough to embed the watermark, the complexity effects on other B and P frames in the same GOP. There are several methods that extract the complexity by using the DCT coefficient's magnitude

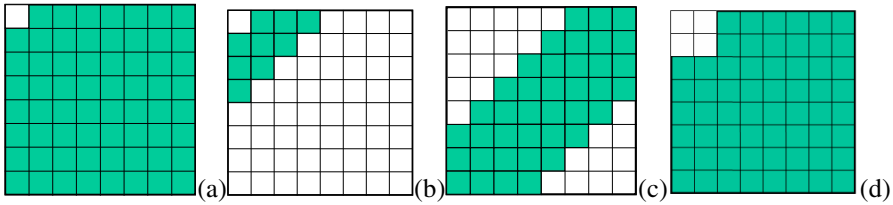


Fig. 4. How to make horizontal difference map

In the proposed algorithm, the method of the complexity calculation is done by the formula 6. The AC components that is near the DC have similar characteristics to DC components

$$Comp = \sum_{i=0}^{63} \left| DCT_{coeff} \right|_i - \sum_{i=0,1,8,9} \left| DCT_{coeff} \right|_i \tag{6}$$

2.2 DCT Based Watermarking

Now, we have to consider whether the watermarking embedded frame is not degraded in the view of human visual system. More than this, we would compare the PSNR with the different DCT coefficients. It is to say, we will choose the proper position in DCT coefficients that is discussed in the previous work.

If we embed the watermarking in the DC components, it could lead the unexpected results of blocking effect. Frank et al proposed the drift compensation to avoid this effect. However, the proposed algorithm could lead more time and buffer because difference between the predictions from frame k and frame k′. The watermarked frame and un-watermarked frame are needed to reduce the block artifacts.

The most important part of this development is no-degrading with the network, so we have to implement the watermarking system with the full compensation of the video quality. So embedding in the DC components is not proper to the new digital broadcasting system.

The next problem is which AC components are better to be embedded. AC coefficients that are composed of 63 different frequency components. Lower parts of AC coefficients show lower frequency parts. Lower parts of AC coefficients are more sensitive to the human visual system and have larger value than higher parts. In the view of these characteristics, embedding position is different from each designer of the watermarking.

We would like to decide the position without degrading the quality maximally. Suppose we have k received signals works, $y_1, y_2 \ldots\ldots, y_k$, carrying the same information bit θ. Each bit θ is multiplied by s_i, i=1~k. With attacks, each $\theta \cdot s_i$ is corrupted with noised

n_1, n_2…,n_k respectively. We assume that the noises are zero mean with variances σ^2_1, σ^2_2...., σ^2_k . We want to determine whether θ is 1 or -1 corresponding to bit one and zero.

$$y_i = \theta \bullet s_i + n_i \quad \text{i=1,…,k} \tag{7}$$

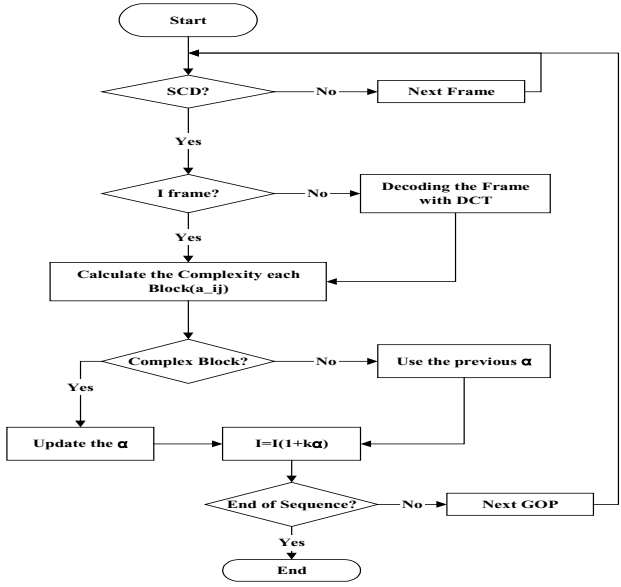


Fig. 5. Flow Chart of the Embedding Watermarking by the Scene Complexity

The test statistics

$$\hat{\theta} = \sum_{i=1}^{k} a_i \left(\frac{y_i}{s_i} \right) \tag{8}$$

$$a_i = \frac{\dfrac{s_i^2}{\sigma_i^2}}{\dfrac{\sum_{j=1}^{k} s_j^2}{\sigma_j^2}} \tag{9}$$

has minimum variance with mean θ if the watermark detector fused received signals with weights a_i.

If we use zero as decision boundary, which means the detected result is bit one if $\hat{\theta} \geq 0$ and the detected result is bit zero otherwise, the probability of watermark detection error is

$$p_e = Q\left[\sqrt{\sum_{i=1}^{k} \frac{s_i^2}{\sigma_i^2}} \right] \tag{10}$$

where Q(x) is the Q-function for the area under the right tail of the Gaussian distribution function. Given the same watermarked image quality, that is, the same total watermark power,

$$C = \sum_{i=1}^{k} s_i^2 \tag{11}$$

We can allocate the watermark power to have a more robust watermark. The watermark power should be allocated to a single DCT coefficient in which the noise variance is the smallest

$$s_r^2 = C\,, \qquad \sigma_r^2 \le \sigma_i^2\,, \qquad \forall i \ne r \tag{12}$$

This allocation scheme is a special case of embedding the watermark bit to the DCT coefficient with the largest signal (watermark) to noise ratio.
In practically, the attacks the watermarked image suffers may not be exactly the same as our assumption. We want to distribute the watermark in multiple DCT coefficients to reduce the risk of allocating all watermark bits in the noisier DCT coefficients.
The embedding watermark step is like that,

First step: Execute the 8*8 DCT transform to get the coefficients.

Second step: Calculate the predicted noise variances of the 63 zigzag ordered DCT coefficients (excluding the DC components). Noises considered are JPEG quantization noise and Gaussian noise. More noised can be taken into account.

Third step: embed each bit of the watermark to its corresponding block in the original frame. The DCT coefficient is raised to embed bit ”1” and is lower to embed bit “0”.

2.3 Extraction and Detection

Watermarking detection procedure is the reverse of the embedding procedure. When watermarked video stream is inputted, we sort the data as a scene change unit with the previous information of video index. After sorting, we decode the data with the first frame that have scene change detected, because such frames is a start frame of the new scene. Watermarked video stream at the scene change position is decoded and inverse quantized using the variable length decoder and inverse quantizer. From this DCT image, watermark is detected. By detecting the watermark, copy protection information is decided.
As watermark is embedded in the DCT domain using spread spectrum method, we use the correlation method that is widely used.

$$sim(X, X^*) = \frac{X \bullet X^*}{\sqrt{X \bullet X^*}} \tag{13}$$

3 Experimental Results

As the M values is increasing, the computation time and suspected scene boundary is more visible than the figure that have a small M value. Final simulation and optimal algorithm is done with the M=13.

The total number of the scene change is 33times. However, because transitions such as a dissolving, fade in and fade out are made in the video sequence, the suspected GOP is increased and we have to consider how to treat these transition.

Table 1 shows the specific scene boundary detection with the motion vector type. First one is detected because of high number of the intra predicted block in B frame. Second one is because of high number of the intra predicted block in P frame. Final one is because of low number of interpolated predicted block and high number of the backward predicted block. The ratio of scene change detection is exactly defined because of the gradual scene change. Total scene change is 33 times and the false detection is 4 and un-detection is 2. Result shows the acceptable result with high motion video data. The music video has very fast movement and sometime scene change is happened within the M values. These constraints make it harder to detect the scene change.

Table 1. Three Type of SCD by Macro Block Type

Num	Gop	Type	FC	FNC	INT	NM	IQ	IPNC	IPC	IPCQ	BNC	BC
708	48	B	17	22	78	0	2	34	116	11	17	26
709	48	B	19	36	76	0	4	48	98	19	17	6
710	48	P	117	1	101	2	0	0	0	0	0	0
711	48	B	66	68	0	0	0	4	1	0	6	0
712	48	B	3	5	0	0	0	1	4	0	58	150
SCD in the B frame because of the Intra Predicted Block												
846	57	B	26	8	93	0	2	17	62	12	12	76
847	57	B	32	23	95	0	4	19	60	19	11	41
848	57	P	26	11	204	8	26	0	0	0	0	0
849	57	B	51	100	0	0	0	1	0	0	5	0
850	57	B	3	3	0	0	0	3	10	0	54	167
SCD in the P frame because of the Intra Predicted Block												
1508	101	P	55	5	155	17	15	0	0	0	0	0
1509	101	B	57	130	0	0	0	5	1	0	1	0
1510	101	B	1	11	0	0	0	4	2	0	105	112
1511	101	P	119	3	6	24	0	0	0	0	0	0
SCD in the B frame because of the Interpolated and Backward Block												

After calculating the complexity, we would like to decide the AC components that are used for the watermarking. If we embed the watermarking in the DC components, we have to compensate the effect of the block artifacts. Even more, if we embed the watermark in any AC coefficients, we can't guarantee the video quality of the water-

marking embedded video. So we have to decide which position is the best for embedding the watermarking. As we have explained, the position determination is done by the variance. We embed the watermark in the low frequency components and middle frequency parts to lower part of each frequency parts. Table 2 shows the variance of the sample I frames. We embedded the watermark in the five AC components. L is in lower and M in the middle frequency domain. By this method we can get the noise protection with invisible watermarking embedding algorithm. To compare this method with the previous method that only use the DC coefficients and fixed AC components, we have simulated several methods in the same broadcasting video data. The result shows the high value of PSNR.

Table 2. Variance of the DCT coefficients

66212	940.85	293.7	134.66	79.653	37.8	29.076	11.644
5639.2	560.66	249.47	188.86	68.625	35.524	19.817	12.007
1816.6	296.09	180.12	119.3	52.046	27.207	20.765	9.3713
783.99	277.51	131.99	92.894	51.707	30.795	17.441	8.3284
531.53	200.31	129.12	91.726	62.1	29.644	16.713	8.9693
355.61	172.95	128.56	79.132	43.07	31.271	16.475	9.5882
329.02	210.27	134.53	100.64	48.244	32.15	13.953	8.897
244.39	160.8	136.94	78.189	54.045	30.326	15.445	9.7745

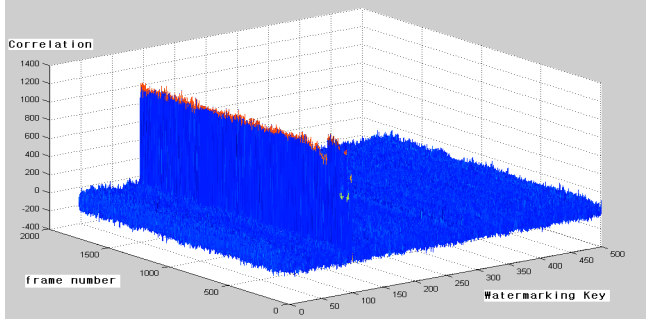


Fig. 6. Watermarking Detection in the full frame

$$PSNR = 10\log_{10}\frac{255^2}{MSE}\ , \tag{14}$$

where MSE is Mean Square Error. It is assumed that pixel intensity lies in the range of 0 to 255.

We have gained the detected watermarking and the correlation value. The high correlation points are shown in the figure.

4 Conclusions

This paper proposed the new watermarking method that is adaptive for the broadcasting system and hard embedded digital broadcasting receiver. We have proposed three factors that were appropriate for the system that don't have enough space and time to process the video stream. One is the scene change detection based on the GOP and another one is VWM based on the scene complexity, and the other is VWM based on the AC variance. The result shows that proposed algorithm's watermarked video have higher quality in most of attacks. Well integration of three factors has achieved the most adaptable video watermarking for the broadcasting system.

Acknowledgement. This work was supported by the Korea Research Foundation Grant (KRF-2003-041-D20470).

References

1. W. A. C. Fernando, C. N. Canagarajah, and D. R. Bull, "Scene change detection algorithms for content-based video indexing and retrieval," Electronics & Communication Engineering Journal, vol.13, No. 3, pp.117-126, June 2001.
2. B.L.Yeo and B.Liu, "Rapid scene analysis on compressed video," IEEE Transactions on Circuit and system for Video Technology, vol. 5, pp. 533-544, 1995.
3. Adan M. Alattar "Detecting and Compressing Dissolve Regions in Video Sequences with a DVI Multimedia Image Compressing Algorithm.", Circuits and Systems, ISCAS '93, IEEE International Symposium, Page(s): 13-16 Vol. 13-6 May 1993.
4. I.J. Cox, M. L. Miller, J. A. Bloom, Digital Watermarking, Morgan Kaufman, U.S.A., 2002.
5. W. A. C. Fernando, C. N. Canagarajah, and D. R. Bull, "Scene change detection algorithms for content-based video indexing and retrieval," Electronics & Communication Engineering Journal, vol. 13, Issue. 3, pp. 117 -126, June 2001.
6. K. Tse, J. Wei, S. Panchanathan, "A scene change detection algorithm for MPEG compressed video sequences" Electrical and Computer Engineering, Canadian conference on, vol. 2, pp. 827 –830, September 1995.
7. J. Huang, Y. Q. Shi and Y. Shi, "Embedding Image Watermarking in DC components," IEEE Transactions on Circuits And Systems for Video Technology, vol 10, no. 6, pp. 974-979, September 2000.
8. Jiwu Huang, Y.Q.Shi, "Embedding strategy for image watermarking in DCT domain," Communications, 1999. APCC/OECC '99. Fifth Asia-Pacific Conference on . and Fourth Optoelectronics and Communications Conference , vol. 2, pp. 981 -984, 18-22 October 1999.
9. Adan M. Alattar "Detecting and Compressing Dissolve Regions in Video Sequences with a DVI Multimedia Image Compressing Algorithm.", Circuits and Systems, ISCAS '93, IEEE International Symposium, Page(s): 13-16 Vol. 13-6 May 1993.
10. Michael Stumpf, "Digital watermarking," in multimedia systems 2002 , University of Southampton, U.S.A.

Distortion-Free of General Information with Edge Enhanced Error Diffusion Halftoning

Byong-Won Hwang[1], Tae-Ha Kang[2], and Tae-Seung Lee[1*]

[1] School of Electronics, Telecommunication and Computer Engineering, Hankuk Aviation University, 200-1, Hwajeon-dong, Deokyang-gu, Koyang-city, Kyonggi-do, 412-791, Korea
bwhwang@mail.hangkong.ac.kr, thestaff@hitel.net
[2] Agency for Defense Development, Yusoung P.O. Box 35, Yusoung-gu, Daejeon-city, Korea
thkang@add.re.kr

Abstract. The error diffusion method is good for reconstructing continuous tones of an image to bilevel tones. However, the reconstruction of edge information by the error diffusion is represented as weak when the power spectrum is analyzed for display error. In this paper, we present an edge enhanced error diffusion method to preprocess original images to achieve an enhancement for the edge information. The preprocessing algorithm consists of two processes. First, the value of difference between the current pixel and the local average of surrounding pixels in the original image is obtained. Second, weighting function is composed by the magnitude and the sign of the local average. To confirm the effect of proposed method, the method is compared with the standard error diffusion and conventional edge enhanced error diffusion methods by measuring various objective measuring criteria including the radially averaged power spectrum density (RAPSD) for display error. The results of comparison demonstrate the superiority of the proposed method over the conventional ones.

Keywords: Computer vision, digital halftoning, edge enhanced error diffusion, differential preprocessing filter

1 Introduction

Image output devices, including printers and faxes, usually have only the two levels of tones or colors in technical and economical reasons. However the devices must output images sawn as natural as possible even if such limitations are imposed. Halftoning is introduced to content with the requirement.

Halftoning is the process to convert continuous-toned image into bilevel-toned one and let see the latter as the former when looked at from a distance. Of many halftoning algorithms studied before, the error diffusion is remarkable for its superior blue-noise property [1]. The error diffusion was proposed by Floyd et al. It distributes the error made at a pixel over the surrounding pixels by quantizing the pixel into

[*] The authors contribute equally to the paper and are listed in alphabetical order.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 281–290, 2004.
© Springer-Verlag Berlin Heidelberg 2004

bilevel tones and using an error diffusion filter that makes the average error for the entire image be zero.

However, the error diffusion filter is designed to retain the average tone of original image, i.e. direct current element frequency, so the degradation of original image for high frequency edge information has to be made [2]. The bilevel-toned image faces contradictory necessities. That is, it has to make the direct current element of display error power spectrum be zero to retain the same average tone to original image, while it has to minimize the error power of high frequency to preserve the original edge information.

The studies conducted to achieve the error diffusion include the methods to modify the error diffusion filter, adaptively adjust filter coefficients to minimize local errors, introduce the property of human visual system (HVS), utilize the characteristics of printers, and so on [3], [4], [5], [6]. Most of all, the edge enhanced error diffusion proposed by Eschbach et al. is remarkable. This method adds multiples of pixel tones to original image in the process of error diffusion to emphasize the edge of original image and get clearer bilevel-toned image. However, the bilevel-toned image converted by the method of Eschbach et al. has some errors at low frequency areas because it uniformly applies the transformation to original image without considering local area characteristics.

This paper studies an improved error diffusion method to maintain edge information with keeping up the enhancement for general information. The heart of the method is a preprocessing filter to reduce the distortions of original image for low and high frequency information. The proposed filter consists of difference value and weighting function. The former is made between a pixel and the local average for the surrounding area in the original image, and the latter use the difference.

The paper hereafter is organized as follows. Section 2 describes the preprocessing filter algorithm proposed by this paper. The performance of the proposed filter is compared with that of the existing edge enhanced methods for various objective measuring criteria including the radial averaged power spectrum density (RAPSD) in Section 3 and is discussed in Section 4. Section 5 finally concludes this paper.

2 Preprocessing Filter Added Edge Enhanced Error Diffusion

The proposed preprocessing filter is designed to maintain general information with keeping up the improvement of edge enhanced error diffusion proposed by Eschbach et al. [7]. The overall error diffusion system is depicted in Fig. 1. The proposed filter is designated by the dotted box and the rest modules are the same to the proposal of Floyd et al. [1]. In the figure, $g(i, j)$ and $h(i, j)$ are the input image and the bilevel-toned image of $I \times J$ samples, respectively. It is assumed that $h(i, j)$ has 0 or 1 and $g(i, j)$ is belong to the range $[0,1]$. $e(i, j)$ is the error generated during quantizing the original tone into 0 or 1.

The proposed filter adds to the quantizer the tone difference of the current pixel to the local area, while the filter of Eschbach et al. multiplies a weighting value directly

to original image and adds the multiplications to the quantizer [7]. The proposed filter is represented by the formulations such as:

$$D_{ij} = G_C - \frac{1}{25}\sum_{k=-2}^{2}\sum_{l=-2}^{2} g(i+k, j+l) \cdot \tag{1}$$

$$G_{ij} = \frac{a_{ij}}{1 + b_{ij} \times \left| D_{ij} \right|} \times sign(D_{ij}) \cdot \tag{2}$$

where, D_{ij} is the difference between the current pixel tone G_C and the local average which is averaged for the 5×5 pixels surrounding to the pixel in the original image. G_{ij} is the weighting function and defined with the magnitude and the sign of D_{ij}. D_{ij} outputs 0 for the even tone distribution of the averaging pixel area, positive values for tones changing like peak, and negative values for valley. When D_{ij} is zero, this means the area is flat in tone distribution and the average tone of the bilevel-toned image will have the similar characteristic to that of the Floyd et al. The coefficient a_{ij} of the weighting function G_{ij} controls the emphasizing level of edge reconstruction and b_{ij} protects edge emphasis from being excessive by steep tones change.

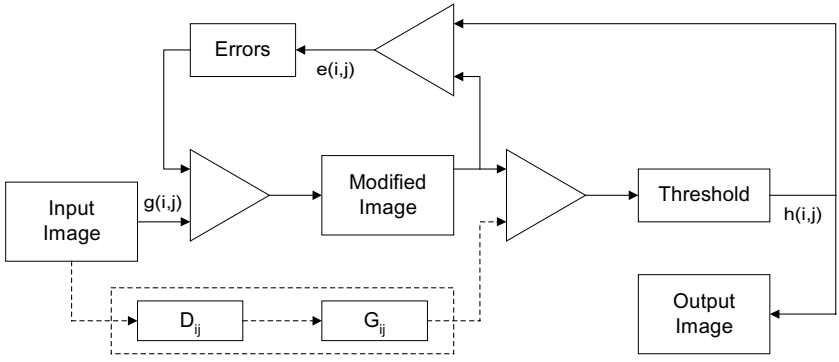


Fig. 1. The edge enhanced error diffusion to which the preprocessing filter is added

3 Evaluation

To evaluate the effect of the proposed edge enhanced preprocessing filter described in Section 2, the two filters of Floyd et al. and Eschbach et al. are compared with the proposed filter for Lena image. This paper adopts three measurement criteria for the objective comparison of them: RAPSD, edge correlation, and local average accordance. In this section, the three measurement criteria are first described, and then the results of comparison for Lena are presented.

3.1 Radially Averaged Power Spectrum Density for Display Error

The RAPSD is a measurement to determine how similar the original image and the bilevel-toned image are to each other [8]. The preferable bilevel-toned image should not have directive biases in pixel pattern and be radially symmetric. This criterion is tested for power spectrum. The power spectrum is defined as $P(f)$ which conducts two-dimensional Fourier transform on bilevel-toned image, squaring of the result, and dividing it by the number of samples. Although $P(f)$ is represented in three-dimension, one-dimensional figure can be presented for effective observation of characteristics by the frequency. The one-dimensional figure is made by partitioning power spectrum into circular rings of width Δ as shown in Fig. 2.

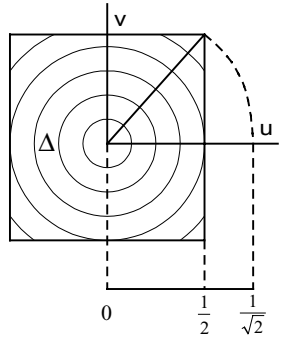


Fig. 2. Partitioning of power spectrum into unit circular rings

This paper constructs the preprocessing filter by utilizing the difference between a pixel and the local average to the surrounding area in the original image. Therefore, for the flat area in tones distribution, the effect of the preprocessing filter is generated little. In this paper, the display error is defined as the difference between the original image and the error diffused bilevel-toned image, and the RAPSD for the display error will be presented in the evaluation.

When the two-dimensional Fourier transform is designated by $\tau[\cdot]$, the power spectrum density is expressed like this:

$$\hat{P}(u,v) = \frac{1}{I \times J} \left| \tau \left[g(i,j) - h(i,j) \right] \right|^2 . \tag{3}$$

The power spectrum is partitioned into circular rings of the uniform width Δ on the basis of center of power spectrum as seen in Fig. 2. In the figure, it is noted that the circular frequency f_r is distant from the center of circular rings by $\Delta_r / \sqrt{2}$. The RAPSD $P_r(f_r)$ is obtained by integrating the power spectrum within the r-th circular ring area and dividing by the number of samples included in the area as follows:

$$P_r(f_r) = \frac{1}{N_r(f_f)} \sum_{i=1}^{N_r(f_f)} \hat{P}(u,v) . \tag{4}$$

where, $N_r(f_r)$ is the number of samples within the r-th circular ring area.

3.2 Edge Correlation

The most important information is in edge area. Therefore, it has objectiveness in quality assessment to measure the correlation for edge area between bilevel-toned and original images. The measuring function C for edge correlation is designed as below:

$$D_g(m,n) = g(i,j) - g(i-m, j-n) \cdot \tag{5}$$

$$D_h(m,n) = g_h(i,j) - g_h(i-m, j-n) \cdot \tag{6}$$

$$C = \sum_{i=0}^{I-1}\sum_{j=0}^{J-1}\left(\sum_{m=-1}^{1}\sum_{n=-1}^{1} W_{mn} D_g(m,n) D_h(m,n)\right) \cdot \tag{7}$$

where, $g_h(i,j)$ is the continuous-toned image restored from the bilevel-toned image by using a 7×7 low-pass filter designed to consider HVS according to observation distance [9]. W_{ij} is the weighting matrix for the horizontal, vertical and diagonal directions. The rate of the diagonal value to the horizontal and vertical values is $1:\sqrt{2}$ and is normalized such that 0.1465 is obtained for the horizontal and vertical directions, and 0.1035 for the diagonal direction. The finally generated function C evaluates the representing performance for edge area of the bilevel-toned image over the original image. Large C means that edge area of bilevel-toned image is consistent with that of original image.

3.3 Local Average Accordance

The performance how much average tone of local area in original image can be preserved is important as well. This performance is evaluated by a function to measure local average accordance between original image and bilevel-toned one. The original image is divided into rectangles of a specific size and the local average of a rectangle is designated as L_{mg}. The bilevel-toned image is reconstructed by using the 7×7 low pass filter mentioned in Section 3.2 and the local average for a rectangle of the reconstructed image is denoted as L_{mh}. The L_{mg} and L_{mh} are formulated like these:

$$L_{mg} = \frac{1}{M^2}\sum_{i=0}^{M-1}\sum_{j=0}^{M-1} g(i,j) \cdot \tag{8}$$

$$L_{mh} = \frac{1}{M^2}\sum_{i=0}^{M-1}\sum_{j=0}^{M-1} g_h(i,j) \cdot \tag{9}$$

where, M^2 is the area to get the local averages. The accordance between the two kinds of local average is defined as follows:

$$A_{Lm} = \frac{1}{\frac{1}{N^2}\sum_{k=0}^{N-1}\sum_{l=0}^{N-1}\left(L_{mg}(k,l) - L_{mh}(k,l)\right)^2} . \tag{10}$$

where, N^2 is the number of the local areas. The large A_{Lm} means that local average of the bilevel-toned image is consistent with that of the original image.

3.4 Experimental Results

The bilevel-toned images generated by the filters of Floyd et al. and Eschbach et al. and the proposed filter are depicted in figures (a), (b) and (c) of Fig. 3, respectively. The figures are cut down from the Lena of original size to consider better resolution of the printed image.

(a) (b) (c)

Fig. 3. Bilevel-toned images generated by the filters of (a) Floyd et al. and (b) Eschbach et al. and (c) the proposed filter

The RAPSDs ($\Delta = 0.004$) for the display errors made between the original images and the bilevel-toned images for Lena are displayed in Fig. 4. In the figure (a) of Fig. 4, the low frequency range of f_r from 0 to 0.3 generates rare RAPSD and the high frequency range from 0.5 to 0.7 high RAPSD. Figure (b) of Fig. 4 reports the RAPSD for the display error by the filter of Eschbach et al. As seen in the figure, the RAPSD for the high frequency range from 0.5 to 0.7 has lower level than that of figure (a). Figure (c) of Fig. 4 shows the RAPSD for the display error by the proposed filter. To obtain the result, $a_{ij} = 2.5$ and $b_{ij} = 0.02$ are used for calculating G_{ij}. The RAPSD for the low frequency range from 0 to 0.2 is low as with figure (a), but over the upper frequency the RAPSD increases until 0.4. In the high frequency range from 0.5 to 0.7, the similar RAPSD to figure (b) is generated.

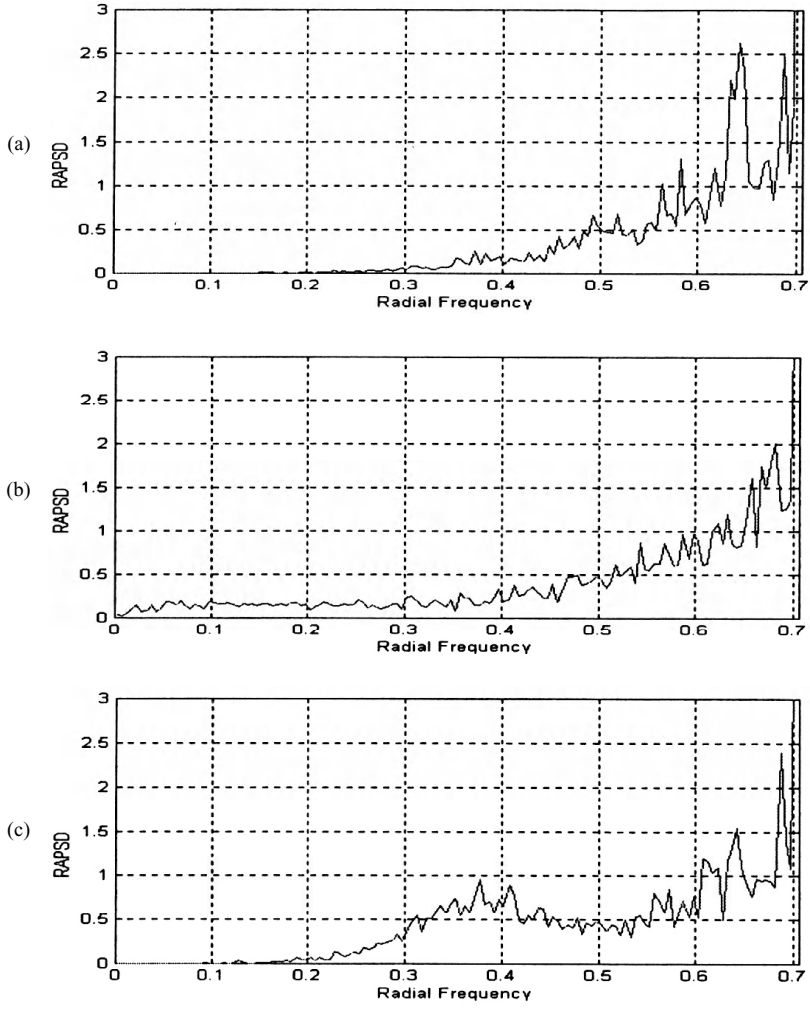


Fig. 4. RAPSD characteristics for the display errors by (a) the filter of Floyd et al.; (b) the filter of Eschbach et al.; (c) the proposed filter

The edge correlation and local average accordance for the bilevel-toned Lena image are recorded in Fig. 5 and Fig. 6, respectively. Fig. 5 presents the edge correlation values as to increasing observation distances for the three filters. In this figure, the values for the filters of Eschbach et al. and the proposed filter are greater than that of Floyd et al. The difference between the two groups decreases with increasing observation distance, but is recognizable when the bilevel-toned image is observed from 10 inches distance. Fig. 6 displays the local average accordance values as to increasing observation distances for all three filters. In this figure, the values for the filters of Floyd et al. and the proposed filter are better than those of Eschbach et al.

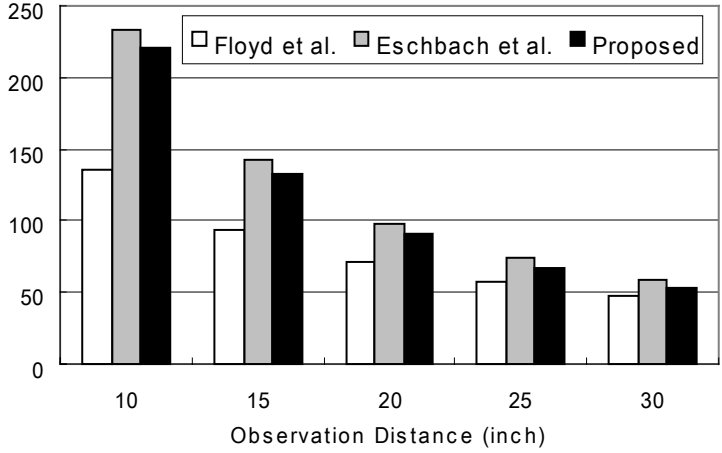


Fig. 5. Comparison of edge correlation values for all the filters

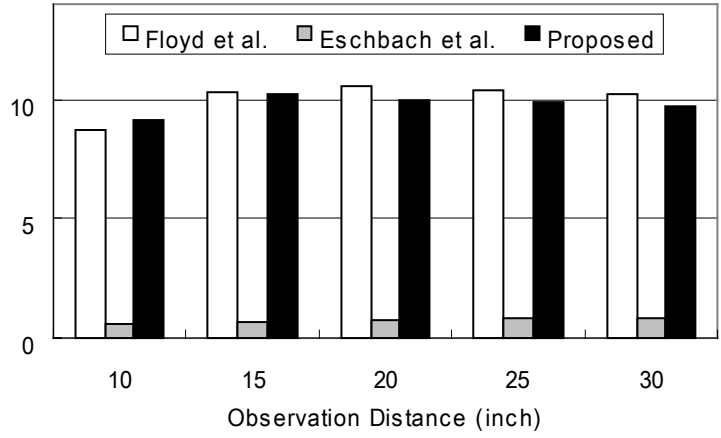


Fig. 6. Comparison of local average accordance values for all the filters

4 Discussion

The results in the aspects of visual, RAPSD for display error, edge correlation and local average accordance confirm an efficient improvement of the proposed filter compared with the filters of Floyd et al. and Eschbach et al. The filter of Eschbach et al. makes bilevel-toned image sharper than that of Floyd et al. does. However, the filter of Eschbach et al. considers little that the negative effect of the edge-enhancing method might cause to damage general information of original image. Compared with

the method of Eschbach et al., the proposed filter can sustain general information as well as enhance edge information.

It is hard to be identified visually how much the filter of Eschbach et al. blurs the general information of Lena, although the visual investigation on Fig. 3 suggests both the filter of Eschbach et al. and the proposed filter improve the edge information of Lena over the bilevel-toned image by the filter of Floyd et al. Such a negative can be found out to compare the RAPSD in the low frequency rage from 0 to 0.2 of figure (b) of Fig. 4 with that of figure (c). It becomes clear when the local average accordance values made by the filters of Eschbach et al. and proposal are examined. As seen in Fig. 6, the distortion of the bilevel-toned image in general information was made seriously for the bilevel-toned Lena image by the filter of Eschbach et al.

The investigation for RAPSD and edge correlation convinces that the proposed filter generates more fine edge information than the filter of Floyd et al. does without losing general information. Figure (c) of Fig. 4 shows that in the high frequency rage from 0.5 to 0.7 the RAPSD of the proposed filter achieves the similar level to that of the filter of Eschbach et al. It is supported by Fig. 5, in which the edge correlation value of proposal at the distance of 10 inches does not show much difference from that of the filter of Eschbach et al., because edge correlation value presents an objective criterion about how much the edge information of original image is preserved into the bilevel-toned image. From the experimental evidences it can be argued that the proposed filter conducts more efficiently edge enhanced error diffusion than the filter of Eschbach et al. does.

5 Conclusion

So far this paper has studied the preprocessing filter emphasizing the edge information of original image based on the standard error diffusion by Floyd et al. with retaining general information. Applying the filter to Lena image and analyzing the bilevel-toned image specified that the sharper bilevel-toned image can be acquired over the error diffusion by Floyd et al. and the more general image over the error diffusion by Eschbach et al. From the experimental results, it can be finally concluded that the proposed filter presents superior properties than the filter of Floyd et al. for the high frequency range that includes most edge information in the original image and that of Eschbach et al. for the low frequency range that includes general information.

References

1. Floyd, R. W., Steinberg, L.: An Adaptive Algorithm for Spatial Greyscale. SID **17** (1976) 75-77
2. Counse, K. R., Roska, T., Chuam L. O.: Image Halftoning with Cellular Neural Networks. IEEE Trans. Circuits and Systems-II **40** (1992) 267-283
3. Jarvis, J., Judice, C., Ninke, W.: A Survey of Techniques for Display of Continuous-Tone Pictures on Bilevel Displays. Comp. Graph. Image Processing **5** (1976) 13-40

4. Wong, P. W.: Adaptive Error Diffusion and Its Application in Multiresolution Rendering. IEEE Trans. Image Processing **5** (1996) 1184-1196
5. Sullivan, J., Miller, R., Pios, G.: Image Halftoning Using a Visual Model in Error Diffusion. J. Opt. Soc. Am. A. **10** (1993) 1714-1724
6. Pappas, T. N., Dong, C. K., Neuhoff, D. L.: Measurement of Printer Parameters for Model-Based Halftoning. Journal of Electronic Imaging **2** (1993) 193-204
7. Eschbach, R., Knox, K.: Error Diffusion Algorithm with Edge Enhancement. J. Opt. Soc. Am. A. **8** (1991) 1884-1850
8. Lau, D. L., Arce, G. R., Gallagher, N. C.: Green-Noise Digital Halftoning. Proceedings of IEE **86** (1998) 2424-2444
9. Pappas, T. N., Neuhoff, D. L.: Least-Squares Model-Based Halftoning. IEEE Trans. on Image Processing **8** (1999) 1102-1116

Enhanced Video Coding with Error Resilience Based on Macroblock Data Manipulation

Tanzeem Muzaffar and Tae-Sun Choi

Mechatronics Department, Kwangju Institute of Science and Technology,
1 Oryong Dong, Puk Gu, Kwangju 500-712, Korea.
{tanzeem, tschoi}@kjist.ac.kr

Abstract. With the rapid growth of video traffic, interest in the coding of video data has increased. Two new techniques are presented to significantly improve video compression ratio, with marginal effect on reconstructed quality. In both these techniques, important data of a macroblock is compressed in one block, while rest of the three data blocks hold difference values in horizontal, vertical and diagonal direction. This results in reduced bitstream size because of low valued data in the three blocks, giving higher compression ratio. These algorithms have an additional advantage that they can be effectively used for error resilience applications with good error handling capacity. For error resilient applications, important data block in a macroblock is transmitted in a secure channel and the remaining three blocks with difference data are sent via lossy channel. In case of error in lossy channel, picture can still be reconstructed with a reasonably good quality using the block transmitted in secure channel that contains important data. Better reconstruction quality is obtained after compression, when used at low bitrates.

1 Introduction

Interest in video compression algorithms is currently motivated by the overgrowing demand of multimedia applications. It is performed on huge amount of video data to increase storage and transmission efficiency. To achieve compression, most video coding techniques exploit large amount of spatial and temporal redundancy present in the highly correlated video data. Temporal redundancy between two successive frames is reduced by using block based motion compensation method, whereas transform coding [1] is used to reduce spatial redundancy i.e. similarities within the image. Quantization process is performed next on transformed coefficients in a lossy manner, to generate high compression ratio at the expense of degradation in reconstructed image quality. For further reduction in size, entropy encoding techniques are used with the expense of increased computation time. This block based coding approach [6-9] is very popular these days and is used in most of commercially available image and video codecs like JPEG, MPEG and H.263 for a variety of applications.

In block based coding approach, a picture (video frame) is arranged in a structure consisting of macroblocks and blocks. A macroblock is a basic building block of this

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 291–300, 2004.
© Springer-Verlag Berlin Heidelberg 2004

coding approach whose construction is same in all the DCT based algorithms. It is composed of 16x16 pixels in the luminance part of a picture. Each macroblock is again split into blocks of 8x8 pixel size. Thus, a macroblock consists of four 8x8 luminance blocks Y. For color video sequences, two spatially correspondent 8x8 chrominance blocks U and V are also added to the macroblock for color information. Since U and V are sub-sampled in both horizontal and vertical directions, there is only one U and V block for every four luminance blocks Y. Block is the smallest unit where the transformation takes place. Each 8x8 block undergoes a DCT transform and results in (transformed) DCT block of the same size. The transformation of blocks into DCT domain exploits spatial redundancy to enhance compression.

The compression scheme providing video services over wireless and other error prone communication networks has led to the development of effective techniques [2][11] to minimize the degradation of video quality caused by errors, in case of data loss. In an error-prone system, error resilience coding is of great importance that can tolerate a limited amount of error during transmission. The simplest method for data error correction is automatic retransmission of requested data, whenever an error is detected at the receiver. Another popular technique to detect and correct data errors is the use of Forward Error Correcting (FEC) codes [5] along with transmitted data. FEC codes are transmitted with each word or packet, thus reducing transmission efficiency. Because of the limited capacity of channels, it is not feasible to provide a complete error free path between source and destination, especially when they are far apart. Therefore, lossy error handling techniques [3-4][10-11] are used to detect errors and minimize its effect on reconstruction rather than correcting it. The most effective scheme for combating channel errors in video applications is to implement multi-layered coding with unequal error protection [4]. A different level of error protection is given to each layer using various error correcting methods, e.g., insertion of error correcting codes and/or automatic retransmission in case of error. In conventional two layer video coding technique for error resilience [13-15] using scalability options, low frequency DCT coefficients that contain most of DCT block information are transmitted in secure channel, whereas rest of the less important data of the block is transmitted in a lossy channel. At the decoder, data is received separately and then combined together for reconstruction. Prediction techniques are used for better reconstruction in case of data loss in a noisy channel.

In this paper, a new concept is introduced to efficiently increase compression ratio of a video sequence data. Two new algorithms have been proposed that can further reduce compressed data size, and generates better reconstructed picture quality at low bitrates. These algorithms have an additional advantage that they can be effectively used for error resilience applications with good error handling capacity. This paper is organized as follows. Two proposed algorithms for efficient video coding with new high compression techniques are explained in detail in section 2. In section 3, experimental results for several video sequences and discussion of results are given. Finally, the paper provides concluding remarks in section 4.

2 Proposed Algorithms

In order to reduce number of bits required to represent a video data, two algorithms are proposed for high compression that uses conventional video codecs like MPEG-2 [6] and H.263. Initially, the picture(video frame) is divided into macroblocks of 16x16 size. These proposed algorithms deal with luminance part of macroblock, which contains four blocks of 8x8 pixels. Basic structure of both the algorithms is similar, whereas difference lies in macroblock arrangement. These algorithms change coefficient positions in a macroblock before DCT, using the concept of wavelet transform. By manipulating values of each block, we can increase compression ratio without sacrificing picture quality significantly. This results in better reconstructed picture quality (SNR) at low bitrates compared to conventional algorithms like MPEG-2. One block is packed with a selected portion of macroblock data, whereas rest of the three blocks holds only difference data in the three directions. Due to the difference blocks, this technique reduces number of bits to encode a macroblock and hence results in high compression of video sequences.

Data of an image separated on its importance within a macroblock can be useful for error resilience applications, which is an additional advantage of these algorithms. Two layer error resilient method can be used for this purpose, in which one layer(base layer) is made highly secure using different error correcting methods where as the other layer(enhancement layer) remains a lossy one. The basic concept is that most of the useful data of a macroblock packed in one block and is transmitted in secure channel(base layer), while rest of the three blocks hold remaining data are sent via lossy channel(enhancement layer). This two-layer error protection scheme is highly resilient to data loss and allows graceful degradation of transmission as the channel encounters increase in channel errors, rather than total corruption of data at the receiver during transmission. Block diagram of proposed coding algorithms is shown in

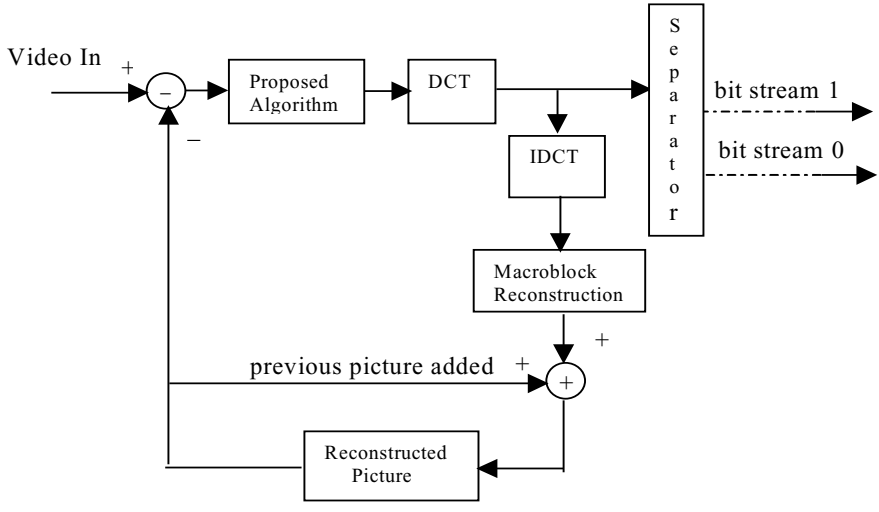


Fig. 1. Block Diagram of the proposed video encoder

2.1 Algorithm 1

In first algorithm, a macroblock is mapped onto four blocks; the first block(A) contains sub-sampled data of a macroblock (with ratio 1:2) in both horizontal and vertical direction. Second (H), third (V) and fourth (D) block in a macroblock gets difference of pixel values with their neighboring sub-sampled values in horizontal, vertical and diagonal directions respectively. Averaging techniques are used to reduce the values of coefficients in these blocks.

Mathematically:

$$
\begin{aligned}
A(i,j) &= a(2i,2j) \\
H(i,j) &= a(2i,2j+1) - [a(2i,2j) + a(2i,2j+2)+1]/2 \\
V(i,j) &= a(2i+1,2j) - [a(2i,2j) + a(2i+2,2j)+1]/2 \\
D(i,j) &= a(2i+1,2j+1) - [a(2i,2j) + a(2i,2j+2) \\
&\quad + a(2i+2,2j) + a(2i+2,2j+2)+2]/4
\end{aligned}
\qquad \ldots (1)
$$

where A, H, V and D are average, horizontal, vertical and diagonal blocks, and a(i,j) is pixel value in the macroblock. Also, *i* and *j* correspond to vertical and horizontal coordinates respectively.

For horizontal block mapping, average of two sub-sampled pixel values is subtracted from the horizontal coefficient in between them. Similarly vertical block coefficients are obtained by taking pixel differences with average of two neighboring sub-sampled values. To obtain diagonal block, we take average of four sub-sampled pixel values surrounding the diagonal pixel, and then take the difference between them. This method compresses the important macroblock information in block A, whereas H, V and D blocks contain only difference pixel values with block A in their respective directions. For reconstruction, the algorithm uses information of block A along with other blocks to get the required image. Figure 2 shows splitting of a macroblock data using algorithm 1.

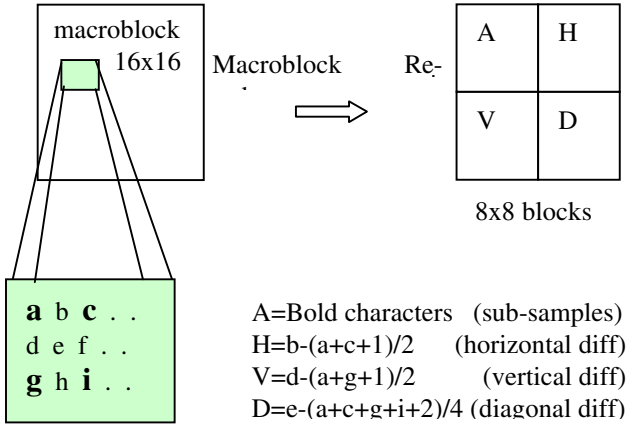


Fig. 2. Splitting of macroblock data into A, H, V and D blocks with Algorithm-1

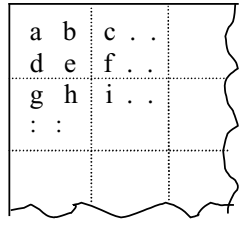
2.2 Algorithm 2

Second algorithm is similar in basic idea of compressing most of the macroblock information in one block. To transform a macroblock into four blocks, this method divides the macroblock into 2x2 windows and then data manipulation is performed on each window by doing simple arithmetic operations. First transformed block(A) contains average values of the corresponding four pixels in 2x2 window, whereas second(H), third(V) and fourth(D) blocks hold values which are average of difference of these 2x2 pixel values in horizontal, vertical and diagonal directions respectively.

Mathematically, it can be represented as:

$$\begin{aligned} A(i,j) &= [a(i,j) + a(i,j+1) + a(i+1,j) + a(i+1,j+1) + 2]/4 \\ H(i,j) &= [a(i,j) - a(i,j+1) + a(i+1,j) - a(i+1,j+1) + 2]/4 \\ V(i,j) &= [a(i,j) + a(i,j+1) - a(i+1,j) - a(i+1,j+1) + 2]/4 \\ D(i,j) &= [a(i,j) - a(i,j+1) - a(i+1,j) + a(i+1,j+1) + 2]/4 \end{aligned} \quad \ldots (2)$$

The rest of the procedure for both these methods is similar to conventional algorithms [7]; i.e. these blocks undergo Discrete Cosine Transform(DCT) and Variable-Length Coding(VLC) for increase in compression ratio. With the proposed coding schemes, resultant data size is considerably reduced due to low-valued data present in H, V and D blocks, thus increasing compression efficiency of the coder. As the manipulated data undergoes transformation (DCT), some loss of data occur which results in slight decrease in reconstructed picture SNR. To have minimum effect on picture quality, these algorithms are applied only to inter and B-pictures in the sequence. The proposed method generates better reconstructed SNR at low bitrates compared to MPEG-2 algorithm. Figure 3 shows a portion of macroblock divided into 2x2 windows for data conversion using algorithm-2.



A=(a+b+d+e+2)/4
H=(a -b+d -e+2)/4
V=(a+b -d -e+2)/4
D=(a -b -d+e+2)/4

Fig. 3. Macroblock data converted into four blocks using Algorithm-2.

2.3 Reconstruction

For proper reconstruction of data compressed through proposed techniques, inverse operation is used on each block. Macroblock data is re-ordered again after undergoing inverse transform (IDCT) to achieve proper results. Inverse operation for algorithm-1 to get macroblock data a(i,j) is computed as:

$$a(2i,2j) = A(i,j) \quad \ldots (3)$$

$$
\begin{aligned}
a(2i,2j+1) &= H(i,j)+[A(i,j)+A(i,j+1)+1]/2 \\
a(2i+1,2j) &= V(i,j)+[A(i,j)+A(i+1,j)+1]/2 \\
a(2i+1,2j+1) &= D(i,j)+[A(i,j)+A(i,j+1)+A(i+1,j)+A(i+1,j+1)+2]/4
\end{aligned}
$$

When algorithm-2 is used, the resultant data after inverse transform is re-ordered using the following equations:

$$
\begin{aligned}
a(2i,2j) &= A(i,j)+ H(i,j)+ V(i,j)+ D(i,j) \\
a(2i,2j+1) &= A(i,j) -H(i,j)+ V(i,j) - D(i,j) \\
a(2i+1,2j) &= A(i,j)+ H(i,j) - V(i,j) - D(i,j) \\
a(2i+1,2j+1) &= A(i,j) - H(i,j) - V(i,j)+ D(i,j)
\end{aligned}
\qquad \ldots (4)
$$

2.4 Error Resilience

When the proposed methods are used for error resilience applications, block A is transmitted in secure channel whereas the other three blocks are sent via lossy channel. In case of error, i.e. loss in H, V and/or D blocks during transmission, data in a lossy channel may be discarded and picture can still be reproduced with a reasonable quality. Reconstruction of picture in case of loss of higher level layer is done using information of block data (A) only, transmitted in secure channel. As long as base layer remains error-free, a satisfactory reconstruction is guaranteed.

3 Experimental Results

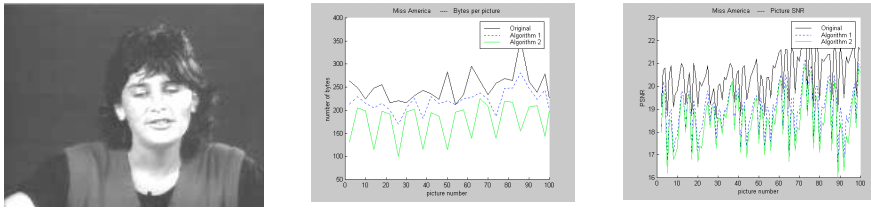
The two algorithms are implemented in software and results are obtained for several video sequences. MPEG-2 video codec is used to test the proposed algorithms with QCIF sequences of Miss America, Carphone and Laboratory. Different quantization parameters (Q = 5, 10, 20) are used in the experiments with N=12 and M=3, where N=distance b/w two I-frames and M=distance b/w two P-frames. Data of a macroblock is manipulated according to the proposed algorithms prior to DCT coding and then coded. At the decoder, macroblock data is re-ordered again before reconstruction, and picture is reconstructed with the help of this data. Experiments show that compression ratio is considerably increased with minimal effect on reconstructed quality. This is because one block is packed with important macroblock data, whereas rest of the three blocks holds only difference data (with low values) in the three directions. These low values in difference blocks eventually result in increased compression of video sequences. High compression performance is observed when they are applied to slow motion pictures. These algorithms are applied only to inter and B-picture transmission, and not to INTRA picture to minimize the effect on reconstructed SNR. Overall, the proposed algorithm generates better reconstructed SNR at low bitrates compared to original MPEG-2 coder.

Tables 1-3 show experimental results of both algorithms in terms of compressed bitstream size and reconstructed SNR for 100 pictures of Miss America, Carphone and Laboratory sequence. Table 1 shows output for quantization parameter Q=5, whereas Table 2 and 3 show results for Q=10 and Q=20 respectively. Comparative

results with original MPEG-2 codec are shown in figures 4 and 5. Figures 4(a) and 5(a) show the reconstructed 100th picture of Miss America and Carphone sequences for Q=10 when algorithm-2 is used. Number of bytes per picture for 100 pictures of these sequences are plotted in Figure 4(b) and 5(b). To clearly show the graphical results of number of bytes/picture obtained from inter and B-picture coding, values are displayed after every 4 pictures (skipping 3 picture values in between them). INTRA picture values are suppressed and not shown in the graphs as the proposed algorithms are not applied to INTRA pictures. Reconstructed SNR for 100 pictures in the sequences are shown in figures 4(c) and 5(c).

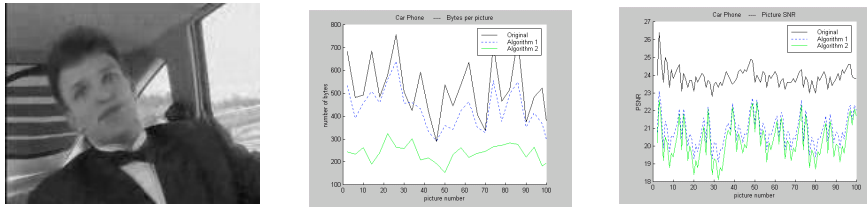
In order to further evaluate the proposed algorithms, Rate-Distortion graph between compression size and reconstructed SNR is shown for Miss America sequence. MPEG-2 is compared with proposed algorithms and shown in figure 6. Compressed bitstream sizes for 100 pictures using different 'Q' are compared with reconstructed SNR of 100th picture. Figure 6(a) plots comparison of MPEG-2 with proposed algorithm-2, whereas 6(b) compares MPEG-2, algorithm-1 and algorithm-2 with compressed bitstream sizes less than 40 Kbytes. These graphs show that algorithm-1 performs better than MPEG-2 at low bitrates, algorithm-2 has better compression capability compared to MPEG-2 even for good quality compression, whereas MPEG-2 outperforms the proposed algorithms only for best quality images.

Figure 7 shows results of improved error resilience with the proposed algorithm compared to conventional method. It shows reconstructed 100th picture of Carphone sequence when error occurs during transmission in a lossy layer, and reconstruction is done using secure layer data only. Figure 7(a) shows reconstruction with conventional method using block DC-coefficients, while figure 7(b) shows reconstruction with the proposed algorithm-2 using block data A only (for Q=10). It can be seen that the proposed algorithm performs good even when used in error-resilient applications.



(a) Recon. 100th pict. using alg-2 (Q=10) (b) Number of bytes per picture (c) Reconstructed picture SNR (dB)

Fig. 4. Results of Miss America sequence suppressing intra picture values



(a) Recon. 100th pict. using alg-2 (Q=10) (b) Number of bytes per picture (c) Reconstructed picture SNR (dB)

Fig. 5. Results of Carphone sequence suppressing intra picture values

Table 1. Simulated results for 100 pictures of QCIF sequence with Q=5 using MPEG-2 (N=12, M=3)

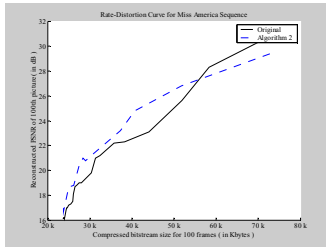
Algorithm used for compression	Miss America Sequence (MPEG compressed)		Carphone Sequence (MPEG compressed)		Laboratory Sequence (MPEG compressed)	
	bytes	SNR(dB)	bytes	SNR(dB)	bytes	SNR(dB)
Compressed bitstream (Original Program)	58382	28.3	112662	28.1	132651	27.7
Algorithm 1 (Inter+B pictures)	56521	25.6	105829	25.4	127509	23.2
Algorithm 2 (Inter+B pictures)	40727	24.8	71213	24.3	65053	21.9

Table 2. Simulated results for 100 pictures of QCIF sequence with Q=10 using MPEG-2 (N=12, M=3)

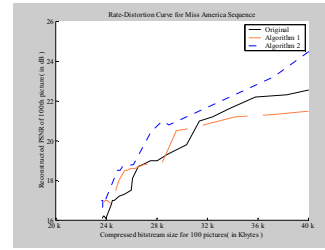
Algorithm used for compression	Miss America Sequence (MPEG compressed)		Carphone Sequence (MPEG compressed)		Laboratory Sequence (MPEG compressed)	
	bytes	SNR(dB)	bytes	SNR(dB)	bytes	SNR(dB)
Compressed bitstream (Original Program)	33758	21.6	66463	23.8	51929	19.8
Algorithm 1 (Inter+B pictures)	31737	20.8	58652	22.0	47230	19.0
Algorithm 2 (Inter+B pictures)	27803	20.6	42159	21.7	34057	18.8

Table 3. Simulated results for 100 pictures of QCIF sequence with Q=20 using MPEG-2 (N=12, M=3)

Algorithm used for compression	Miss America Sequence (MPEG compressed)		Carphone Sequence (MPEG compressed)		Laboratory Sequence (MPEG compressed)	
	bytes	SNR(dB)	bytes	SNR(dB)	bytes	SNR(dB)
Compressed bitstream (Original Program)	26012	17.5	39459	19.8	30802	15.9
Algorithm 1 (Inter+B pictures)	24425	16.8	34149	19.0	27751	15.5
Algorithm 2 (Inter+B pictures)	23745	16.7	31095	18.9	26227	15.5



(a) Comparison of original & proposed Algorithm-2



(b) Comparison of original, Algorithm-1 and Algorithm-2 with bitstream size between 20 Kbytes and 40 Kbytes.

Fig. 6. Rate-Distortion graph – SNR vs. compressed bitstream size of Miss America Sequence

(a) Conventional method - Reconstructed 100th picture of Carphone sequence using DC-coefficient

(b) Proposed method 2 - Reconstructed 100th picture of Carphone sequence using block A data only(Q=10)

Fig. 7. Reconstructed 100th picture of Carphone sequence in case of error

4 Conclusions

Two techniques for video compression are presented and implemented on a computer. Important data of a macroblock is packed in one block while remaining less important data (with small values) in macroblock is put in other three blocks. Better reconstructed picture quality(SNR) is achieved at low bitrates using the proposed algorithms, compared to original MPEG-2 coder. For error-resilient applications, the important data block is transmitted in a lossless channel, whereas the remaining data of three blocks is sent via a noisy channel. In case of data loss, picture is not corrupted completely, but can still be reconstructed with a reasonable quality using block data received from secure channel only. Execution time and complexity of algorithms is negligibly increased as only simple arithmetic operations are used, making it suitable for real time applications.

Acknowledgement. This work was supported by the Korea Research Foundation Grant (KRF-2003-041-D20470)

References

1. C.A. Gonzales, L.Allman, T. McCarthy, P. Wendt, "DCT coding for motion video storage using adaptive arithmetic coding", Signal Processing: Image Communication 2, vol. 2, No. 2, 1990.
2. Je-Cheon Yoon, S. H. Lee, "Reduction of blocking effect in transform domain using neural network", IEEE Tencon'97 Conference, December 1997.
3. M.R. Frater, J.F. Arnold, J.Zhang, "MPEG-2 video error resilience experiments: The importance considering the impact of the system layer", Signal Processing, Image Communication, 1997.
4. S.Aign, K.Fazel, "Temporal and spatial error concealment techniques for hierarchical MPEG-2 video codec", IEEE International Conference on Communication, vol. 3, 1995.
5. K. Rao, J. Hwang, "Techniques and Standards for image, video and audio coding", Prentice Hall Publishing Company, 1996.
6. "MPEG Software Simulation Group (MSSG)", http://www.mpeg.org/MPEG/MSSG.
7. ISO/IEC/JCT1 13818-2, "Generic coding of moving pictures and associated audio", March 1994.
8. T. Sikora, "MPEG Digital video coding standards" IEEE Signal Processing Magazine, vol. 14, September 1997.
9. K. Konstantinides, C.T. Chen; T.C. Chen; H. Cheng; F.C. Jeng, "Design of an MPEG-2 video codec", IEEE Signal Processing Magazine, vol 19, July 2002.
10. W. S. Lee, M. R. Pickering, M. R. Frater & John Arnold, "Error Resilience in Video and Multiplexing Layers for Very Low Bitrate Video Coding Systems", IEEE Journal on Selected areas in Comm, vol. 15, No. 9, December 1997.
11. Y, J, Chiu, "A perceptual based video coder for error resilience", IEEE Data Compression Conference (DCC), March 1999.
12. R.C. Chang, T.T. Lu, "A scalable video compression technique based on wavelet transform and MPEG coding", IEEE Transaction on Consumer Electronics, vol. 45, No.3, August 1999.
13. M. Ghanbari, "Two-layer coding of video signals for VBR networks", IEEE Transaction on Selected Areas on Communication, vol. 7, no 5, June 1989.
14. C. Lee, D. Lee, J. Park, Y. Kim, "A new two layer video compression scheme for multiple applications", IEEE Transaction on Consumer Electronics, vol. 38, no 3, August 1992.
15. D. Wilson, M. Ghanbari, "Optimization of two layer SNR scalability", ICASSP Proceedings, April 1997.

Filtering of Colored Noise for Signal Enhancement

Myung Eui Lee[1] and Pyung Soo Kim[2]

[1] School of Information Technology,
Korea Univ. of Tech. & Edu., Chonan, 330-708, Korea
[2] Mobile Platform Lab, Digital Media R&D Center,
Samsung Electronics Co., Ltd, Suwon City, 442-742, Korea
Phone : +82-31-200-4635, Fax : +82-31-200-3147
kimps@samsung.com

Abstract. This paper suggests an enhancement approach for signal corrupted by additive colored noise signal. The well known FIR structure filter is adopted in order to obtain the noise-suppressed estimate of the desired signal. It is shown that the suggested approach has the quick estimation ability for desired signal. It is also shown that the estimate of the desired signal is separated from the additive colored noise signal when the additive colored noise signal is nearly constant on the window. In addition, when the additive colored noise signal itself is treated as an additional desired signal that should be estimated, its estimate is shown to be separated from the state term for the original desired signal. Via numerical simulations on a military signal, the performance of the suggested approach is evaluated by the comparison with that of the existing Kalman filtering approach.

1 Introduction

In real world, there are many signals such as audio signals, military signals, and biomedical signals, etc. There are also many applications using these signals such as voice communication systems and speech recognition systems for audio signals, global positioning systems and inertial navigation systems for military signals, and electroencephalogram analysis systems for biomedical signals. However, in many applications, these signals are often corrupted by additive noise signals such as white noises or colored noises. Therefore, to enhance the desired signal corrupted by the additive noise signal, the statistical signal processing for noise suppression must be required. Several attempts to use the Kalman filtering to enhance the desired signal corrupted by the colored noise signal have been made [1]-[3]. In these approaches, the desired signal and the additive colored noise signal are represented in state space signal models in order to utilize the Kalman filtering algorithm. However, since Kalman filter is an infinite impulse response (IIR) structure utilizing all information on the infinite interval as time goes and has a recursive formulation, the Kalman filtering approach may show poor performance and even divergence phenomenon for temporary modeling uncertainties and round-off errors [4], [5].

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 301–310, 2004.
© Springer-Verlag Berlin Heidelberg 2004

It has been a general rule of thumb in statistical signal processing areas that a finite impulse response (FIR) structure, which utilizes only the information on the finite interval, is often used in many cases instead of the IIR structure, since the former has a bounded input bounded output (BIBO) stability, robustness to temporary modeling uncertainties and round-off errors, and a linear phase property when necessary [6], [7]. In addition, the FIR structure can avoid long processing time due to the large data sets in case of the IIR structure when time increases. Therefore, in the current paper, an alternative approach to enhance the desired signal corrupted by the colored noise signal is suggested using the well known FIR structure filter in [8]. This FIR structure filter processes linearly measurements on the most recent finite interval called the window, has a batch formulation, doesn't require *a priori* statistics information of the initial state and has the properties of unbiasedness, minimum variance and efficiency.

In the suggested FIR filtering approach, it is shown that the exact desired signal is obtained within finite time when there are no exitation and measurement noises in actual incoming signal although the filter is designed with consideration of them. This indicates that the suggested approach has the quick estimation ability for the desired signal. In addition, this quick estimation property cannot be obtained from the Kalman filtering approach in [1]-[3]. Therefore, when the desired signal corrupted by the colored noise signal varies relatively quickly, the suggested approach will give a better estimate compared with existing Kalman filtering approaches. In addition, it is shown that the estimate of the desired signal is separated from the additive colored noise signal when the additive colored noise signal is nearly constant on the window. Moreover, when the additive colored noise signal itself is treated as an additional desired signal that should be estimated, its estimate is shown to be separated from the state term for the original desired signal. These separating estimation properties cannot be also obtained from the Kalman filtering approach in [1]-[3]. Via numerical simulations on the military signal used in inertial navigation systems, these good inherent properties of the suggested approach are verified. In addition, numerical simulations show that the performance of the suggested approach is better than that of the Kalman filtering approach in [1]-[3].

2 FIR Filtering for Signal Enhancement

The main task of the current work is a filter design to enhance the desired signal corrupted by the colored noise signal as well as the measurement noise using only the measured incoming signal $z(i)$. The desired signal, the additive colored noise signal, and measured incoming signal can be represented in the following state-space signal model as shown in [1]-[3]:

$$x_d(i+1) = A_d x_d(i) + G_d w_d(i), \tag{1}$$

$$z(i) = C_d x_d(i) + C_n x_n(i) + v(i), \tag{2}$$

$$x_n(i+1) = A_n x_n(i) + G_n w_n(i) \tag{3}$$

where $x_d(i)$ and $x_n(i)$ are state vectors for the desired signal and the additive colored noise signal, respectively. The measurement noise $v(i)$ is a zero-mean white noise with covariance R. Excitation noises $w_d(i)$ and $w_n(i)$ are zero-mean white noise with covariance Q_d and Q_n, respectively. These excitation noises and are mutually uncorrelated and also mutually uncorrelated with $v(i)$.

Augmenting (1)-(3), the following state space signal model is obtained:

$$x(i+1) = Ax(i) + Gw(i), \tag{4}$$

$$z(i) = Cx(i) + v(i) \tag{5}$$

where the state and exitation noise vectors and parameter matrices are

$$x(i) = \begin{bmatrix} x_d(i) \\ x_n(i) \end{bmatrix}, w(i) = \begin{bmatrix} w_d(i) \\ w_n(i) \end{bmatrix}, A = \begin{bmatrix} A_d & 0 \\ 0 & A_n \end{bmatrix}, G = \begin{bmatrix} G_d & 0 \\ 0 & G_n \end{bmatrix}, \ C = \begin{bmatrix} C_d \, C_n \end{bmatrix}.$$

Noise $w(i)$ is zero-mean white and mutually uncorrelated with $v(i)$. The covariance of $w(i)$ is the diagonal matrix Q whose elements are Q_d and Q_n.

To obtain the noise-suppressed estimate of the desired signal, the FIR structure filter in [8] is applied to the state-space signal model (4) and (5). This FIR structure filter processes linealy the only finite measurements on the most recent window $[i-M(\triangleq i_M), i]$ and discards the past measurements outside the window for the estimate at the present time i. In addition, this FIR structure filter has a batch formulation, doesn't require *a priori* statistics information of the initial state and has the properties of unbiasedness, minimum variance and efficiency.

For the state space signal model (4) and (5), the FIR structure filter is defined by the following simple batch form:

$$\hat{x}(i) \ = \ HZ(i). \tag{6}$$

When $\{A, C\}$ is observable and $M \geq p + q - 1$, the filter coefficient matrix H can be obtained from [8]. The measurements $Z(i)$ on the most recent window $[i_M, i]$ can be represented in the following regression form from the desired signal model (1) and (2):

$$Z(i) = L_d x_d(i_M) + \bar{C}_n X_n(i) + \bar{G}_d W_d(i) + V(i) \tag{7}$$

where

$$L_d \triangleq \begin{bmatrix} C_d \\ C_d A_d \\ \vdots \\ C_d A_d^M \end{bmatrix}, \ \bar{C}_n \triangleq \begin{bmatrix} C_n & 0 & \cdots & 0 & 0 \\ 0 & C_n & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & C_n \end{bmatrix},$$

$$\bar{G}_d \triangleq \begin{bmatrix} 0 & 0 & \cdots & 0 & 0 \\ C_d G_d & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ C_d A_d^{M-1} G_d & C_d A_d^{M-2} G_d & \cdots & C_d G_d & 0 \end{bmatrix}.$$

and

$$Z(i) \triangleq [z(i_M)^T \;\; z(i_M+1)^T \;\; \cdots \;\; z(i)^T]^T$$

and $X_n(i)$, $W_d(i)$, $V(i)$ have the same form as $Z(i)$.

Since each row of the filter coefficient matrix H is the subfilter for each individual state, the estimates for the desired signal and the additive colored noise signal are obtained simultaneously as follows:

$$\hat{x}(i) = \begin{bmatrix} \hat{x}_d(i) \\ \hat{x}_n(i) \end{bmatrix} = HZ(i) = \begin{bmatrix} H_d \\ H_n \end{bmatrix} Z(i) \tag{8}$$

where H_d and H_n are given by the first p rows and the last q rows of the filter coefficient matrix H. Thus, the estimate $\hat{x}_d(i)$ for the desired signal is given by

$$\hat{x}_d(i) = H_d Z(i).$$

The noise-suppressed estimate $\hat{x}_d(i)$ for the desired signal processes the finite measurements on the most recent window linearly, doesn't require *a priori* statistics information of the window initial state and has the properties of unbiasedness, minimum variance and efficiency. Note that the Kalman filter used in [1]-[3] does not have above properties unless the mean and covariance of the initial state is completely known. In addition, due to the FIR structure and the batch formulation, the suggested FIR filtering approach guarantees the BIBO stability, and may have the robustness to temporary modeling uncertainties and to round-off errors, while the Kalman filtering approach might be sensitive for these situations.

3 Inherent Properties of Suggested Approach

In this section, it will shown that the suggested FIR filtering approach has some good inherent properties such as the quick estimation property and the separating estimation property.

As shown in [8], the FIR filter used in this paper provides the exact desired signal within finite time when there are no exitation and measurement noises, i.e., $w_d(i) = w_n(i) = v(i) = 0$ in (1)-(3), although their covariances Q_d, Q_n, R in the filter design are nonzero. This property indicates that the suggested approach has the quick estimation ability for the desired signal. In addition, this quick estimation property cannot be obtained from the Kalman filtering approach in [1]-[3]. Therefore, when the desired signal corrupted by the colored noise signal varies relatively quickly, the suggested approach will give a better estimate compared with existing Kalman filtering approaches.

Using this quick estimation property, when $w_d(i) = w_n(i) = v(i) = 0$, equations (1), (3) and (7) give the following:

$$\begin{aligned} \hat{x}(i) &= H\big[L_d x_d(i_M) + \bar{C}_n X_n(i)\big] \\ &= H\big[L_d x_d(i_M) + \bar{C}_n \bar{A}_n x_n(i_M)\big] \\ &= \begin{bmatrix} H_d \\ H_n \end{bmatrix} \begin{bmatrix} L_d & L_n \end{bmatrix} \begin{bmatrix} A_d^{-M} & 0 \\ 0 & A_n^{-M} \end{bmatrix} x(i) \end{aligned}$$

where

$$\bar{A}_n = \begin{bmatrix} I \\ A_n \\ \vdots \\ A_n^M \end{bmatrix}, \quad L_n = \begin{bmatrix} C_n \\ C_n A_n \\ \vdots \\ C_n A_n^M \end{bmatrix}$$

Therefore, the following matrix equalities are always satisfied:

$$H_d L_d = A_d^M,\ H_n L_n = A_n^M,\ H_d L_n = H_n L_d = 0 \tag{9}$$

which will be used in following theorems.

It is shown that the estimate for the desired signal is separated from the additive colored noise signal when the additive colored noise signal is nearly constant on the window.

Theorem 1. When the additive colored noise signal nearly constant on the window $[i_M, i]$*, the estimate* $\hat{x}_d(i)$ *in (8) for the desired signal is separated from the additive colored noise signal.*

Proof : When the noise signal $x_n(i)$ is nearly constant as $\bar{x}_n$ on $[i_M, i]$, the finite measurements $Z(i)$ (7) can be represented in the following regression form:

$$Z(i)\big|\{x_n(\cdot) = \bar{x}_n \text{ on } [i_M, i]\} = L_d x_d(i_M) + \bar{C}_n \bar{A}_n \bar{x}_n + \bar{G}_d W_d(i) + V(i). \tag{10}$$

Then, the estimate $\hat{x}_d(i)$ for the desired signal is derived from (8)-(10) as

$$\begin{aligned} \hat{x}_d(i) &= H_d\big[L_d x_d(i_M) + \bar{C}_n \bar{A}_n \bar{x}_n + \bar{G}_d W_d(i) + V(i)\big] \\ &= H_d L_d x_d(i_M) + H_d L_n \bar{x}_n + H_d\big[\bar{G}_d W_d(i) + V(i)\big] \\ &= A_d^M x_d(i_M) + H_d\big[\bar{G}_d W_d(i) + V(i)\big] \end{aligned}$$

which does not include the additive colored noise signal term. ∎

As mentioned previously, when the additive colored noise signal itself is treated as an additional desired signal, it should be estimated. In this case, the estimate for the additive colored noise signal is shown to be separated from the state term for the desired signal.

Theorem 2. The estimate $\hat{x}_n(i)$ in (8) for the additive colored noise signal is separated from the state term for the desired signal.

Proof: The estimate $\hat{x}_n(i)$ for the additive colored noise signal is derived from (7)-(9) as

$$\begin{aligned}\hat{x}_n(i) &= H_n\big[L_d x_d(i_M) + \bar{C}_n X_n(i) + \bar{G}_d W_d(i) + V(i)\big] \\ &= H_n L_d x_d(i_M) + H_n \bar{C}_n X_n(i) + H_n\big[\bar{G}_d W_d(i) + V(i)\big] \\ &= H_n \bar{C}_n X_n(i) + H_n\big[\bar{G}_d W_d(i) + V(i)\big]\end{aligned}$$

which does not include the state term for the desired signal. ∎

As the quick estimation property, these separating estimation properties in Theorem 1 and 2 cannot be also obtained from the Kalman filtering approach in [1]-[3]. These good inherent properties of the suggested approach are verified via numerical simulations in the next section.

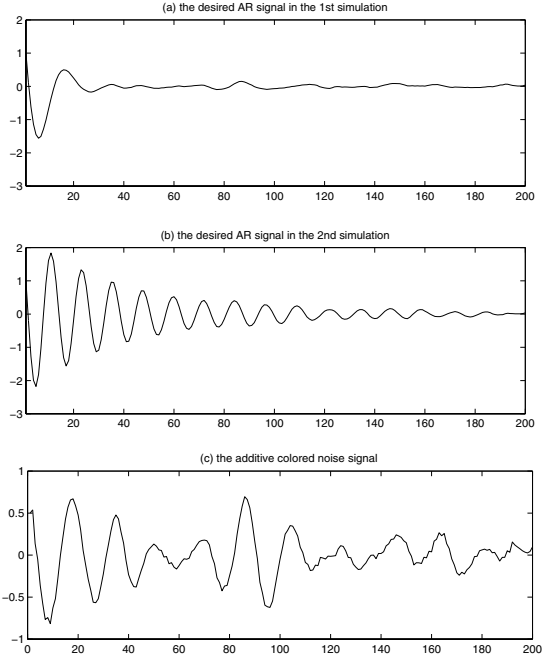


Fig. 1. Test signals

4 Simulations

In order to evaluate performance of the suggested FIR filtering approach, the spacecraft attitude tracking scheme with the gyroscope as a sensor is considered [2], which has been often used in inertial navigation systems. The main objective of the spacecraft attitude tracking scheme is to enhance and track the spacecraft drift signal corrupted by the additive colored noise signal as well as the measurement noise using only the measured incoming signal from the gyroscope. Thus, the spacecraft drift signal becomes the desired signal. In addition, since to find out the cause of corrupted drift signal is required, it is necessary to estimate the additive colored noise signal.

There are two simulations for different two spacecraft drift signals which vary with second order AR model as follows:

$$A_d = \begin{bmatrix} 0 & 1 \\ a_{d_1} & a_{d_2} \end{bmatrix}, \quad G_d^T = C_d = \begin{bmatrix} 0 & 1 \end{bmatrix}.$$

In the first simulation, the spacecraft drift signal is assumed to vary relatively slowly as $a_{d_2} = 1.7$ and $a_{d_1} = -0.8$. In the second simulation, the spacecraft drift signal is assumed to vary relatively quickly as $a_{d_2} = 1.7$ and $a_{d_1} = -0.95$. For two simulations, the additive colored noise signal is assumed to vary relatively quickly as following the third order AR model:

$$A_n = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -0.6 & 0.2 & 1.2 \end{bmatrix}, \quad G_n^T = C_n = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}.$$

The design parameters for the FIR filtering are taken as follows. The window length is taken as $M = 20$. The covariances of exitation and measurement noises are taken as $Q_d = 0.01^2$, $Q_n = 0.04^2$ and $R = 0.02^2$.

The performance of the suggested approach is evaluated by the comparison with the Kalman filtering approach in [1]-[3]. To make a clearer comparison, fifty Monte Carlo runs are performed and each single run lasts for 200 samples. Test signals used in one of fifty runs are plotted in Figure 1 to show characteristics of the spacecraft drift signal as the desired signal and the additive colored noise signal. As shown in Figure 1 (a), the spacecraft drift signal in the 1st simulation varies relatively slowly. As shown in Figure 1 (b), the spacecraft drift signal in the 2nd simulation varies relatively quickly. For these spacecraft drift signals, the additive colored noise signal in two simulations varies relatively quickly as shown in Figure 1 (c). Root-mean-square (RMS) errors of estimates for these spacecraft drift and additive colored noise signals are shown in Figure 2-5.

For the estimate of the spacecraft drift signal which varies relatively slowly, the performance of the suggested approach is shown to be similar to that of the Kalman filtering approach, as shown in Figure 2 (a) and 3 (a). However, for the estimate of the spacecraft drift signal which varies relatively quickly, the suggested approach outperforms remarkably the Kalman filtering approach, as shown in Figure 4 (a) and 5 (a). Note that Theorem 1 can be the theoretical

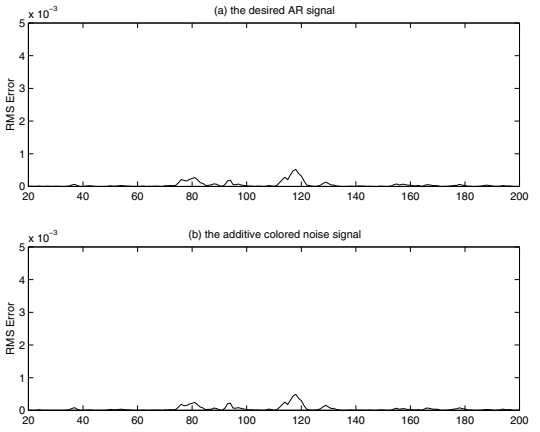


Fig. 2. Result of suggested FIR filtering based approach : 1st simulation

background of these results. Therefore, it can be known that, when the desired signal corrupted by the colored noise signal varies relatively quickly, the suggested approach gives a better estimate compared with the Kalman filtering approach in [1]-[3].

For the estimate of the additive colored noise signal which varies relatively quickly, the performance of the suggested approach is shown to be better to that of the Kalman filtering approach for two all simulations, as shown in Figure 2 (b), 3 (b), 4 (b) and 5 (b). Especially, in the 2nd simulation where the spacecraft drift signal varies relatively quickly, the performance difference between two approaches is remarkable as shown in Figure 4 (b) and 5 (b) although the additive colored noise signal is same as one in the 1st simulation. This indicates that the

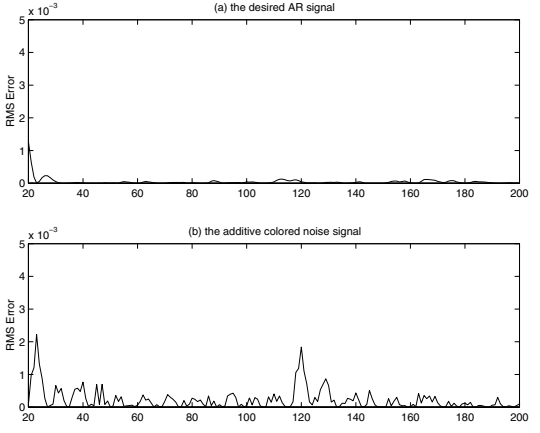


Fig. 3. Result of Kalman filtering based approach : 1st simulation

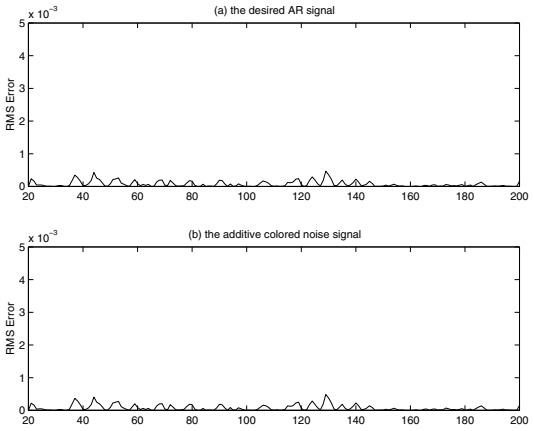


Fig. 4. Result of suggested FIR filtering based approach : 2nd simulation

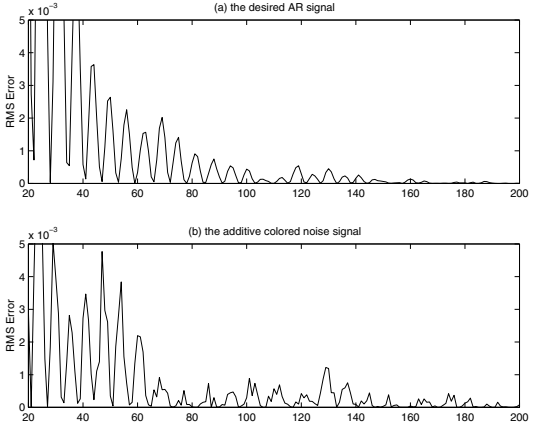


Fig. 5. Result of Kalman filtering based approach : 2nd simulation

estimate of the additive colored noise signal in the suggested approach might be less affected by the spacecraft drift signal than the case of in the Kalman filtering approach. Note that Theorem 3 can be the theoretical background of these results. Therefore, it can be known that, when the additive colored noise signal itself is treated as an additional desired signal that should be estimated, its estimate in the suggested approach is less affected by the original desired signal than the case of the Kalman filtering approach in [1]-[3].

5 Concluding Remarks

This paper has suggested the FIR filtering of colored noise for signal enhancement. The suggested approach provides the quick estimation ability for desired

signal, which will give a better estimate compared with existing ones when the desired signal corrupted by the colored noise signal varies relatively quickly. It is shown that the estimate of the desired signal is separated from the additive colored noise signal when the additive colored noise signal is nearly constant on the window. In addition, when the additive colored noise signal itself is treated as an additional desired signal that should be estimated, its estimate is shown to be separated from the state term for the original desired signal. Moreover, the suggested approach guarantees the BIBO stability, and may have the robustness to temporary modeling uncertainties and to round-off errors, while the Kalman filtering approach might be sensitive for these situations. Via numerical simulations on the military signal, good inherent properties of the suggested approach are verified. In addition, numerical simulations show that the performance of the suggested approach is better than that of the Kalman filtering approach.

References

1. Gibson, J.D., Koo, B., Gray, S.D.: Filtering of colored noise for speech enhancement and coding. IEEE Trans. Acous., Speech, Signal Processing Vol.39. (1991) 1732–1742
2. Jiang, H., Yang, W.Q., Yang, Y.T.: State space modeling of random drift rate in high-precision gyro. IEEE Trans. Aerosp. Electron. Syst. Vol.32. (1996) 1138–1143
3. Gannot, S., Burshtein, D., Weinstein, E.: Iterative and sequential Kalman filter-based speech enhancement algorithm. IEEE Trans. Speech and Audio Processing Vol.6. (1998) 373–385
4. Fitzgerald, R.J.: Divergence of the Kalman filter. IEEE Trans. Automat. Contr. Vol.16. (1971) 736–747
5. Xie, L., Soh, Y.C., de Souza, C.E.: Robust Kalman filtering for uncertain discrete-time systems. IEEE Trans. Automat. Contr. Vol.39 (1994) 1310–1313
6. Schweppe, F.: Uncertain Dynamic Systems. Englewood Cliffs, NJ:Prentice-Hall (1973)
7. Oppenheim, A., Schafer, R.: Digital Signal Processing. Englewood Cliffs, NJ:Prentice-Hall (1975)
8. Kwon, W.H., Kim, P.S., Han, S.H.: A receding horizon unbiased FIR filter for discrete-time state space models. Automatica Vol.38 (2002) 545–551

Model-Based Human Motion Tracking and Behavior Recognition Using Hierarchical Finite State Automata

Jihun Park[1], Sunghun Park[2], and J.K. Aggarwal[3]

[1] Department of Computer Engineering
Hongik University, Seoul, Korea
jhpark@hongik.ac.kr
[2] Department of Management Information Systems
Myongji University, Seoul, Korea
shpark@mju.ac.kr
[3] Department of Electrical and Computer Engineering
The University of Texas at Austin, Austin, TX 78712
aggarwaljk@mail.utexas.edu

Abstract. The generation of motion of an articulated body for computer animation is an expensive and time-consuming task. Recognition of human actions and interactions is important to video annotation, automated surveillance, and content-based video retrieval. This paper presents a new model-based human-intervention-free approach to articulated body motion tracking and recognition of human interaction using static-background monocular video sequences. This paper presents two major applications based on basic motion tracking: motion capture and human behavior recognition.

To determine a human body configuration in a scene, a 3D human body model is postulated and projected on a 2D projection plane to overlap with the foreground image silhouette. We convert the human model body overlapping problem into a parameter optimization problem to avoid the kinematic singularity problem. Unlike other methods, our body tracking does not need any user intervention. A cost function is used to estimate the degree of the overlapping between the foreground input image silhouette and a projected 3D model body silhouette. The configuration the best overlap with the foreground of the image least overlap with the background is sought. The overlapping is computed using computational geometry by converting a set of pixels from the image domain to a polygon in the 2D projection plane domain.

We recognize human interaction motion using hierarchical finite state automata (FA). The model motion data we get from tracking is analyzed to get various states and events in terms of feet, torso, and hands by a low-level behavior recognition model. The recognition model represents human behaviors as sequences of states that classify the configuration of individual body parts in space and time. To overcome the exponential growth of the number of states that usually occurs in a single-level FA, we present a new hierarchical FA that abstracts states and events from motion data at three levels: the low-level FA analyzes body parts only,

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 311–320, 2004.
© Springer-Verlag Berlin Heidelberg 2004

the middle-level FAs recognize motion and the high-level FAs analyze a human interaction. Motion tracking results and behavior recognition from video sequences are very encouraging.

1 Introduction and Previous Work

Analysis of video data is important due to the rapid increase in the volume of content recorded in the form of video. Recognition of human interaction in video is important to video annotation, automated surveillance, and content-based video retrieval. Recognizing human interactions is a challenging task because it involves segmentation and tracking of articulated human body parts at low level and recognition of semantics in behavior at a higher level. This paper presents a model-based approach to motion tracking and recognition of human interaction in static-background monocular video sequences. Our motion tracking is based on computational geometry and forward kinematics to avoid the singularity problem [1], while our behavior recognition is based on hierarchical deterministic finite state automata (DFA) to abstract motion data in hierarchies. This paper is an extension to our previous papers[2,3,4]. This paper differs in hierarchically handling behavior recognition, and not relying on a distance map in overlapping computation, compensating recognition in [4].

While many others use stochastic sampling for model-based motion tracking, our method is purely dependent on parameter optimization. We convert the human motion-tracking problem into a parameter optimization problem. A cost function for parameter optimization is used to compute the degree of the overlapping between the foreground input image silhouette and a projected 3D model body silhouette. The overlapping is computed using computational geometry by converting a set of pixels from the image domain to a polygon in the real projection plane domain. One parameter optimization solves the model body configuration problem for a set of input images, and we compute the body configuration from each image frame. From a set of model body configuration data, we build the motion data. The model motion that we get from tracking is analyzed to get various states and events. From the motion data, we extract events that occur during the motion as well as changes in the configuration. We abstract numerical model body motion data into a sequence of state-change data. We recognize the motion of a single body from a sequence of state changes. We recognize interactive motion from each model body state changes. The above approach is hierarchical.

We may classify human motion analysis methods according to the recognition algorithms used: either stochastic algorithms such as hidden Markov models (HMM), or deterministic algorithms such as finite state automata (FA). If the uncertainty in image can be effectively resolved by model-based methods at the low level, then we can use deterministic methods for motion interaction recognition. In this case, the reliable overlapping between the model body and image data is useful. Many approaches have been proposed for behavior recognition using various methods, including HMM, FA, and context-free grammar. Oliver

et al. [5] presented a coupled hidden Markov model for gross-level human interactions. Hongeng et al. [6] proposed probabilistic finite state automata(FA) for gross-level human interactions. Their system utilizes user-defined hierarchical multiple scenarios of human interaction. Hong et al. [7] proposed a DFA for detailed-level recognition of human gestures. Wada et al. [8] used nondeterministic finite state automata (NFA) using *state product space*. They preferred NFA to HMM [8] because NFA provides transparent state transition information whereas HMM's state transition is *hidden* to the user.

Our motion capturing and recognition framework successfully captures and recognizes five common human interactions: walking (i.e., approaching, departing), pushing, pointing, kicking, and handshaking. The major contributions of our work are as follows: (1) To overcome the occlusion problem, we have developed an overlapping area computation based on computational geometry, that automatically detects the initial configuration of the model body.[2,3] (2) To overcome the singularity problem encountered in inverse kinematics, we convert the problem to a forward kinematics based parameter optimization problem.[2] (3) Our motion tracking is fully automatic without any user intervention. (4) To overcome the problem of exponential growth of motion states encountered in single-level FA, we have developed a hierarchical FA to analyze body part motion and recognize motions and interactions. Motion capture results from color video sequences are very encouraging.

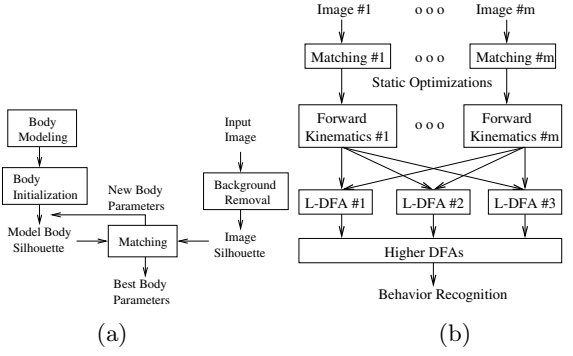


Fig. 1. Process of determining the best matching model body configuration for a single image(a), and a sequence of image matching, motion/behavior recognition(b).

2 Overview of Our System

Our system is designed for model-based human motion tracking and recognition of human interactions in surveillance videos. Figure 1(a) shows a matching process (i.e. computing the overlapping between the foreground input image and projected model body silhouette) given an input image. A 3D model body

is built up. Model body joint angles and displacements are determined, and the body is projected on a 2D projection plane. On the 2D projection plane, we get a model body silhouette. The boxes in Figure 1 represent computational processes. The *matching* process uses static parameter optimization [9], which modifies the model body parameters and checks the resulting level of matching between the image silhouette and the model body silhouette for a single image. When the best matching model body configuration is found for a single image, then the process is done for that image; thus, for n input images, we run the matching computation n times to get n set of tracked motion data.

Figure 1(b) shows the sequence of matching process tasks. When the matching computation is completed using static optimization, we have a model body configuration for each image. Then we run forward kinematics to determine the kinematic parameters such as hand position and foot position for each image. The kinematic parameters of the fitted model body form the motion data, which is then analyzed by a recognition model. We propose a *hierarchical* deterministic finite state automata (DFA) as the recognition model. The hierarchical DFA is composed of low-level DFAs to abstract numerical motion data and analyze the motion data with respect to the feet, torso, and hands. The low-level DFAs independently represent the individual body-part poses as discrete states and the body part motion as transitions between the states. The body part motion recognition results from the low-level DFA are fed into middle-level DFAs for the recognition of a whole body motion, and then fed into higher-level DFAs to analyze the interactive motion behavior between two persons.

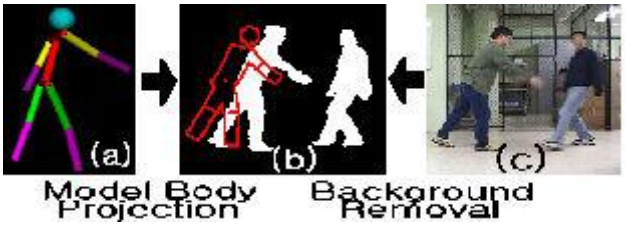


Fig. 2. 3D Model body(a), overlapping between background removed image and projected model body(b), and input image(c)

In this section, we present our optimization cost function to find the best overlap. The main cost function is very similar to our previous functions[2,3, 4] except that we no longer use a distance map. As shown in Figure 2(a), the body is modeled as a configuration of nine cylinders and one sphere. These are projected onto a 2D real projection plane. A sphere represents the head, while the rest of the model body is modeled using cylinders of various radii and lengths. Currently, we use only nine 1-DOF (1 degree-of-freedom) joints plus body displacement DOF. These are our control variables for the parameter optimization and the cost function.

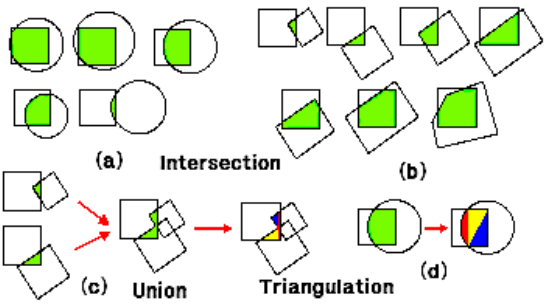
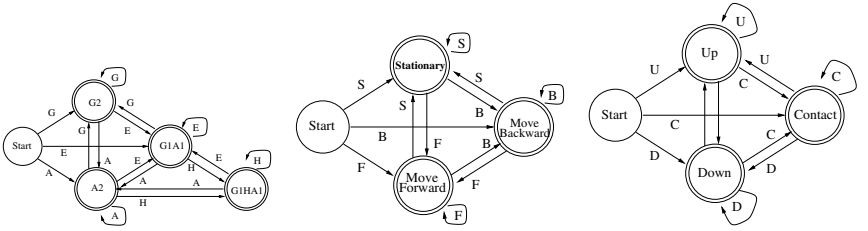


Fig. 3. Five possible cases of a pixel(square) partially occluded by a model head(a), seven possible cases of a pixel(square) partially occluded by a polygon body(b), union of intersected area, then triangulation for a polygon body(c), and union and intersection computation for a model head(d)



(a) Foot movement analysis. (b) Body center movement analysis. (c) Hand movement analysis.

Fig. 4. Lower level finite state automatas for recognizing body part motion.

While a model silhouette is computed by a 3D human model projection, an image silhouette has been converted from the 2D integer pixel domain to a real domain such that the resulting image silhouette becomes a jagged-edge polygon with only horizontal and vertical edges. We compute the polygon intersection between the image silhouette and the model silhouette. We found the best fitting configuration using the GRG2[9] optimization package with the best matching overlap between the model body silhouette and the foreground of the input image. Figure 2(b) shows the initial state of searching for the best overlapping configuration, given the first frame image of the video sequence of Figure 6(e). As can be seen in Figure 2(b), the initial joint angle values of the model body for parameter optimization are arbitrary. This shows that our initial model body configuration detection for the first image frame is automatic. We know the center of the foreground of an input image, and matching is done using optimization. We know by how much the model part covers/misses the foreground/background of an input image. From the foreground image, we can compute how tall and thick the human is because we are given side view of the input images. By finding the best overlapping, we automatically find the best body configuration. The background removal process is presented in our previous papers[2,3,4].

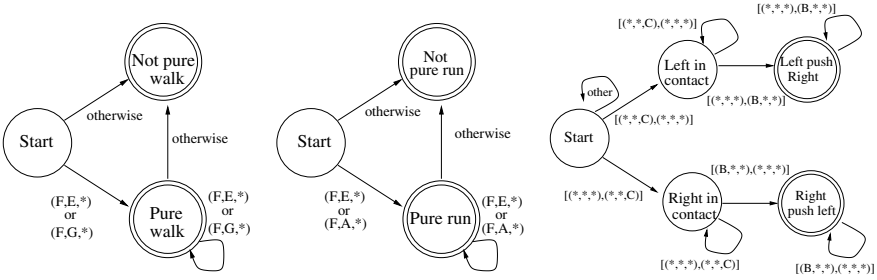
Figure 3 shows 12 possible overlapping cases in which either a model head or polygon-shaped body, generated after 3D model body projection, is overlapping with a pixel. In the figure, a circle represents a projected head outline, and an irregular polygon represents a body part, while a square represents a pixel. The union of these irregular-shaped objects results in a projected model body silhouette. After union computation, a triangulation process is needed to compute the area of the unioned irregular shaped object. Because our cost function of the parameter optimization works only in the real number domain, we cannot work on a purely pixel-based integer number domain cost function. Thus we have to compute the pixel overlapping area to eliminate an integer cost function.

3 Hierarchical Deterministic Finite State Automata for Behavior Recognition

We model human behavior as sequences of state changes that represent the configuration and movement of individual body parts (i.e., legs, torso, and hands) in spatio-temporal space. We employ a *sequence analyzer* and an *event detector* that are quite similar to those of [8]. However, our sequence analyzer is a DFA that tracks status changes, unlike the nondeterministic finite state automata of [8], while the event detector allows state transition. Our DFAs are quite unique because we have hierarchical layers of sequence analysis. The use of hierarchical DFAs reduces the exponentially large number of states to be handled in behavior recognition. Each DFA consists of a finite set of states(Q), an initial state (q^0), a finite set of events ($\sum$), a state transition function (δ), and a finite set of final states (F). It is represented by $(Q, q^0, \sum, \delta, F)$. Each state q^i in the sequence $(q^0, q^1, \cdots, q^n)$ corresponds to a frame. To handle every possible situation, our low-level sequence analyzers are of the form $({}^p_m Q, {}^p_m q^0, {}^p_m \sum, {}^p_m \delta, {}^p_m F)$, where $p, p = 1, 2, 3$, is an index for body parts, index number one for body center, index number two for feet, index number three for hands, and m is an index for each person in the scene. ${}^1_2 q^i \in {}^1_2 Q$ means ${}^1_2 q^i$ is a state of sequence index number i, of a *second* person in the scene, of body part index *one*, the body center. The event detector detects events or status changes while reading model body motion data obtained from a sequence of parameter optimization. Events are determined from model motion data. To detect a specific event, it is necessary to check a sequence of motion.

We employ DFAs on three levels: A separate low-level DFA is employed for each body part: body center (torso), feet, and hands. Each low-level DFA considers all possible states for its body part (hand, torso or feet), independent of the rest of the body. Low-level DFA input is a set of numerical data that is abstracted and converted into a sequence of status changes. We allow only four states for feet: both feet on the ground, both feet in the air, one foot in the air while the other is on the ground, and one foot in the *high* air while the other is on the ground. (This state is to recognize violent foot motion.) The walking motion is less dependent on arms or hand movement. Figure 4(a) shows a DFA to analyze feet status. At the start state, there are only four transitions possible

because we classify every situation into one of four states. State G2 means a status in which both feet are on the ground, state A2 means both feet are in the air, state G1A1 means one foot is on the ground while the other is in the air, and state G1HA1 means one foot is in the *high* air. The state transition occurs when a condition in the motion configuration is satisfied, and is denoted as G, A, E, or H. Similarly, we define three body center (torso) states: stationary, moving forward, and moving backward, denoted as S, F, and B, respectively. Figure 4(b) shows a DFA to analyze body center status. Three states are defined for hands: both hands down (D), at least one hand raised (U), and at least one hand in contact with another person (C). Figure 4(c) shows a DFA to analyze the status of the hands. We do not differentiate between hands.



(a) Recognizing pure walking (b) Recognizing pure running (c) High-level DFA.

Fig. 5. Middle-level finite state automatas for recognizing motion of a single person(a,b), and high-level DFA for recognizing interactive (pushing) motion of two persons(c).

A middle-level DFA that takes low level DFA state changes as input is used to recognize a single body. Figure 5(a,b) shows middle-level DFAs for recognizing walking and running. We consider a tuple of states for a middle-level DFA, $({}^1_m q^i, {}^2_m q^i, {}^3_m q^i)$, a token made of low-level state transitions of a model body of index number m, to recognize its motion. * means any possible input. Each DFA recognizes a specific motion only. DFAs can be modified for user's video content retrieval request. State changes at low level DFA are fed into a middle level DFA, which determines a single body motion status. The results from the middle-level DFA is a higher-level motion description such as "stand still" or "kick" of a single person. The outputs from the middle-level DFA as well as low-level DFAs are fed into the high-level DFA, which analyzes the interactive motion behavior between model bodies.

We now explain how the high-level DFA works. To recognize an interactive motion between two persons, we need a tuple of states, $[({}^1_1 q^i, {}^2_1 q^i, {}^3_1 q^i), ({}^1_2 q^i, {}^2_2 q^i, {}^3_2 q^i)]$. The tuple consists of states of a left and right person's body part status. The higher-level DFA(s) states recognize behavior based on lower-level sequence analyzers, thirteen lower-level states to abstract motion data. Rather than using an exponentially increasing number of entire states, we

focus on a subset of all states. Each tuple we fed into a higher-level DFA corresponds to one of an exponentially large number of states. For a higher-level DFA, the number of states can be relatively small. For each person in a scene, there are approximately 36 possible states because we use three or four states for each of the three body parts. If there are two persons involved in an interaction, we would need to make a DFA of minimum 1296 states to handle all possible motion states. Generally, we need $|{}^1_1Q| \times |{}^2_1Q| \times |{}^3_1Q| \times |{}^1_2Q| \times |{}^2_2Q| \times |{}^3_2Q|$ states for an interaction of two persons, where $|Q|$ is the number of states in Q. It is plain that this exponential growth will quickly become intractable. Rather than generating 1,296 states and designing state transitions, we design three or four states to recognize each motion of a body part, totaling 13 states for any number of persons. As a result, we need to design a higher level DFA to recognize behavior based on lower-level sequence analyzers, plus 13 lower-level states to abstract motion data rather than 1,296 states and state transition designs.

4 Experimental Results

Our system was used to recognize five 2D-based human motions: walking (i.e., approaching, departing), pushing, kicking, pointing, and handshaking. Figure 6(f,h) shows two persons shaking hands in front of a static background. The red line shows the union of every model body part. As long as there is no heavy occlusion between the subjects, motion tracking is satisfactory because of geometric union computation in handling occlusion. The motion tracking is excellent, as shown in Figure 6. Figure 6(g,i) shows a walking (departing) motion. After motion tracking, we get two sets of motion data, one for each person appearing in the scene. The raw motion data is abstracted and converted to a sequence of states for each body part.

Figure 6(a) shows a pushing motion. The right person approaches and pushes the left person. As a result of being pushed, the left person moves backward. This is an example of a scene in which a cause (pushing) of a person resulted in a change (move backward) in the other person. This interaction can be only recognized by checking all 1296 states, including many states that are not directly related to a pushing motion. SSSSSSSSSSSSSNNSSNNNNNNNNNNNNNNNNNNN is a sequence of body center states of the left person in the scene, where **S** represents "stand still" state, and **N** means "move negative." DUUUUCCCCCCCUUUUDDDDDDDDDDDDDDDDDDDDDDD is a sequence of hand states of the right person in the scene, where **D** represents "the both hands down" state, **U** means "any hand up", and **C** means "hand(s) in contact." Two sequences of the input tuple consist of these state change sequences. From the input tuple, we can easily recognize behavior: "right person contacted the left person, and the left person has moved to negative direction," meaning the right person pushed away from the left person. We need at least four states to recognize a pushing motion: one representing the contact state by a pushing person, and the other representing the backwards movement of the pushed person, for both persons in the scene. A complicated query is of the form "a person approached the other, and pushed away," which would require all three levels of DFA-based *sequence analysis.*

Fig. 6. The subject, with the model figure superimposed, shown over a pushing motion (a), a walking (approaching) motion (b), a kicking motion (c), a pointing motion (d), a pushing motion (e), a hand-shaking motion (f), a walking (departing) motion (g), a hand-shaking motion (h), and a walking (departing) motion (i).

5 Conclusion

In this paper, we presented a new approach to human motion capture and its behavior analysis using hierarchical DFAs. The model based-method at the image processing level uses a 3D human body model and parameter optimization techniques to achieve refined segmentation and tracking of the moving humans. Use of the model body in human motion tracking allows us to take advantage of the knowledge of the human body inheritance in the model, making the system more robust. The motion tracking results from video sequences are very encouraging, although it performs best on side views. The output data from model-based human tracking enables us to recognize human behavior in the input scene. Rather than using an exponentially increasing number of entire states, we focus on a subset of all states. Our recognition framework successfully recognizes various human interactions between two persons, although our current motion states cannot cover all human motion.

Acknowledgements. This research was supported by the 2004 Hongik University Academic Research Support Fund. We thank Ms. Debi Prather for proofreading of this paper.

References

1. Morris, D., Rehg, J.: Singularity analysis for articulated object tracking. In: Computer Vision and Pattern Recognition. (1998)
2. Park, J., Park, S., Aggarwal, J.K.: Human motion tracking by combining view-based and model-based methods for monocular video sequences. Lecture Notes in Computer Science (2003 International Conference on Computational Science and Its Applications) **2669** (2003)
3. Park, J., Park, S., Aggarwal, J.K.: Model-based human motion capture from monocular video sequences. Lecture Notes in Computer Science (ISCIS 2003) **2869** (2003)
4. Park, S., Park, J., Aggarwal, J.K.: Video retrieval of human interactions using model-based motion tracking and multi-layer finite state automata. Lecture Notes in Computer Science (2003 Intl. Conf. on Image and Video Retrieval) **2728** (2003)
5. Oliver, N.M., Rosario, B., Pentland, A.P.: A Bayesian computer vision system for modeling human interactions. IEEE Trans. Pattern Analysis and Machine Intelligence **22** (2000) 831–843
6. Hongeng, S., Bremond, F., Nevatia, R.: Representation and optimal recognition of human activities. In: IEEE Conf. on Computer Vision and Pattern Recognition. Volume 1. (2000) 818–825
7. Hong, P., Turk, M., Huang, T.S.: Gesture modeling and recognition using finite state machines. In: IEEE Conf. on Face and Gesture Recognition. (2000)
8. Wada, T., Matsuyama, T.: Appearance based behavior recognition by event driven slective attention. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Orlando, FL (1998) 759–764
9. Lasdon, L., Waren, A.: GRG2 User's Guide. (1989)

Effective Digital Watermarking Algorithm by Contour Detection

Won-Hyuck Choi, Hye-jin Shim, and Jung-Sun Kim[1]

School of Electronics, Telecommunication and Computer Engineering, Hankuk Aviation University, 200-1, Hwajeon-dong, Deokyang-gu, Koyang-city, Kyonggi-do, 412-791, Korea
rbooo@korea.com,{hjshim,jskim}@mail.hangkong.ac.kr

Abstract. In this paper, it proposes digital watermarking algorithm that can protect copy right of an image by using contour extract techniques. In an image, the area where the number of contour has very complicated images can insert a large amount of watermark, which results in satisfaction of firmness and non-visibility of watermark. In addition, the information of watermark converts a copy of owner's information to ASCII code and by using converted ASCII code to binary code of 8 bytes, it does not concern similarity between watermarks in case of their distinction. Also, it suggests watermarking algorithm that does not require an original image and information of an inserted watermark.

1 Introduction

It is possible for us to get a large amount of information from the Internet by the popularization of computer and the development of communication techniques. In the Internet, countless information in digital form is interacting to share and communicate. However, there are accompanying problems to utilize information from Internet, and the most important issue is about copyright. As a response of the arising issue, many studies are in process to resolve illegal copying and alteration of digital work and to protect copyright.

Digital Watermarking, one of resolutions of the issue, is one of the ways to protect copyright of a copy owner [1]. Digital watermarking is a way to prove the existence of hidden information for a creation by inserting secrets that is only known by the copy owner. In general, the major requirements for watermarking technique are introduce below [2].

i. The inserted watermark should maintain the quality of work and invisible to other people.
ii. For image process techniques like JPEG (Joint Photography Experts Group) or filtering, watermark should be firm.
iii. The extracted watermark data should be clear to distinguish the copy owner.
iv. It should be able to find the same watermark by comparing watermark treated to two images.

[1] The corresponding author will reply to any question and problem from this paper

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 321–328, 2004.
© Springer-Verlag Berlin Heidelberg 2004

In the paper, it suggests digital watermarking method that can protect copyright of digital image. The area to insert watermark in an original image is decided by contour detecting method. In an image, the area where has number of contour has very complicated images so can insert much amount of watermark, which results satisfaction of firmness and non-visibility of watermark. The contour detecting method sets a point of weight by doing Zig-Zag scanning to an image and inserts watermark to a part where increasing rate is more than a critical value.

Also, the information of watermark converts a copy owner's information to ASCII code and does not calculate similarity between watermarks in case of their distinction because of using converted ASCII code to binary code of 8 bytes. In addition, it suggests watermarking algorithm that does not require an original image and information of an inserted watermark.

The constitution of the paper is like below. In chapter 2 Related studies examines concept of digital watermarking and existing method of study. Chapter 3 introduces digital watermarking algorithm by contour detection. And in the last chapter, chapter 4 draws conclusion and suggests further plan for the study.

2 Background

Until the present, digital watermarking method is proposed for protection of copyright. The previously proposed methods divide an original image into a block with a unified size and change the block. In the middle of process, information for watermark is annexed to a specified part of the converted block and its outcome becomes inverted. Moreover, detection method of watermark reverses its performance to insert watermark and acknowledges the watermark if the detected watermark satisfies cohesion [5].

Such watermarking technologies are classified according to practical application technique; first is treating data in Spatial Domain and second is treating data in Frequency Domain [2][3].

The spatial technique is a method of analyzing data, like an image, in spatial point of view and dispersing the data that is going to be inserted in given space not to distinguish it easily. In general, it is a method to use watermark for minute change of screen pixel. It is easy to insert watermark in this method; however, it is not effective to manage image process like JPEG or filtering [6].

The frequency domain technique has various merits than watermarking with spatial analysis. The frequency utilized method converts multimedia data to analogue signal of frequency element and also converts watermark identically, which is going to be inserted, as an analogue signal - then insert [7].

Ways to convert data are FFT: Fast Fourier Transform, DCT: Discrete Cosine Transform, DWT: Discrete Wavelet Transform, etc and they are methods to convert an image [8]. These methods are difficult to delete already inserted watermark because inserting watermark modulus, which is data, is distributed throughout the whole domain of original image.

3 Digital Watermarking Algorithm by Contour Detection

For a method to satisfy invisibility and firmness of digital watermark, the inserting position to insert maximum size watermark in an original image is decided by contour detection method, and information of copy owner is converted as binary information on the detected image and inserted.

3.1 Method of Contour Detection

In this thesis, a way to decide inserting domain for watermark through the characteristic of contour is suggested. A block diagram for contour detection is shown in fig. 1. This method detects a part that has the largest change in light and shade from an original image.

Contour detection uses only numbers of light and shade so RGB image, which has to go through complicated calculation, is converted to Gray level image. In order to insert watermark, using Gaussian mask is more effective to get rid of unnecessary noise from an image than simply detecting minute contour in an image. Gaussian mask is often used to show background and major image and detect a part where has great difference in light and shade.

Once noise is gone, a part where has great difference in light and shade is detected with contour by utilizing Sobel filter. The contour detected image data decides a position to insert watermark because the contour converged spot is difficult to detect much change according to visual characteristic of neighboring picture element.

3.1.1 Method to Decide the Domain for Watermark Insertion

The contour detected image decides a domain of watermark insertion. An image that completed process of contour detection is inputted and it is converted to either 1 or 0 according to numbers of contour and background. These numbers is used to make a curve line to indicate increment of contour through zig-zag scan that JPEG compression algorithm uses.

A section that has the highest increment becomes a domain to insert watermark in the contour increment curve. The algorithm for deciding a domain to insert watermark by contour detection method in an image is like below.

[Algorithm 1] Algorithm for deciding a domain to insert watermark
Input: Contour detected image
Output: A domain for inserting watermark

```
Embedding_Watermark() {
   Int S[x][y], i, j, x, y, k, Sum[x*y];
   // Convert 0 and 1 according to contour
   for(x=0; x<X_Max; x++) {
      for(y=0; y < Y_Max; y++) {
         if (S[x][y] > Middle_Value) S[X][Y] = 0;
         else S[x][y] = 1;
      }
   }
   // 8 *8 Being blocked
```

```
   for(i=0; i<X_Max; i+=8) {
      for(j=0; j < Y_Max; j+=8) {
         // Accumulate the sum based on  zig-zag scan in a block
         Zig_Zag Scan;
         Sum[x*y] + = Sum[x][Y]
      }
   }
   // Draw a curve.
   for(i=0; i<X_Max; i+=8) line(0, k, k, sum[k]);
}
```

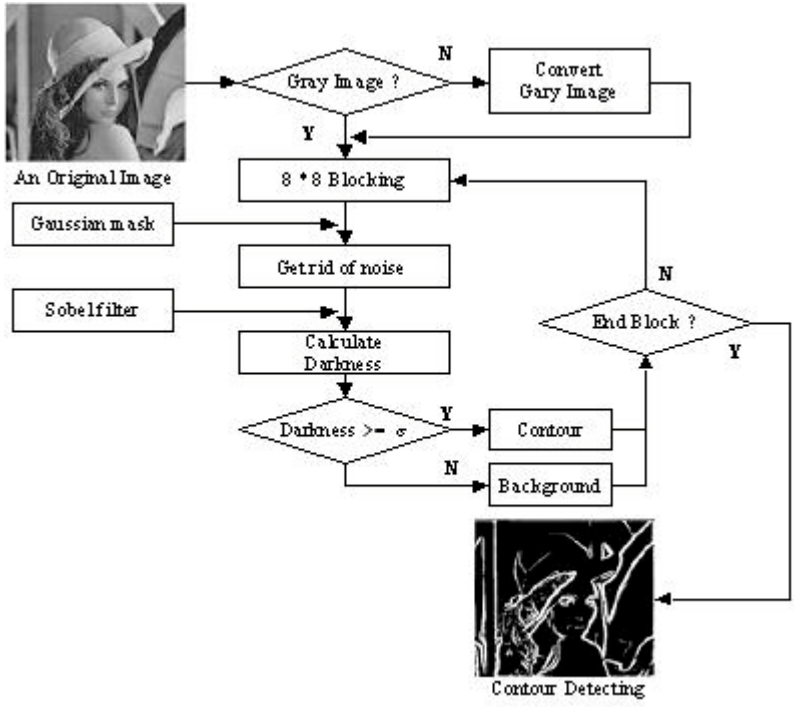


Fig. 1. A block diagram for contour detection

3.2 Algorithm for Watermark Formation with Information of Copy Owner

In this study, possibility of watermark process is examined for detection of watermark and information of a copy owner is used to confirm one's copyright about an image. Assembled information such as name, mailing address, contact number, date of image production of copy owner is called as W, and it is converted to binary code and creates B_W.

When W is converted to binary code, a copy owner's information is converted to ASCII code and then one letter like Fig. 2 is changed to a code with 8bit.

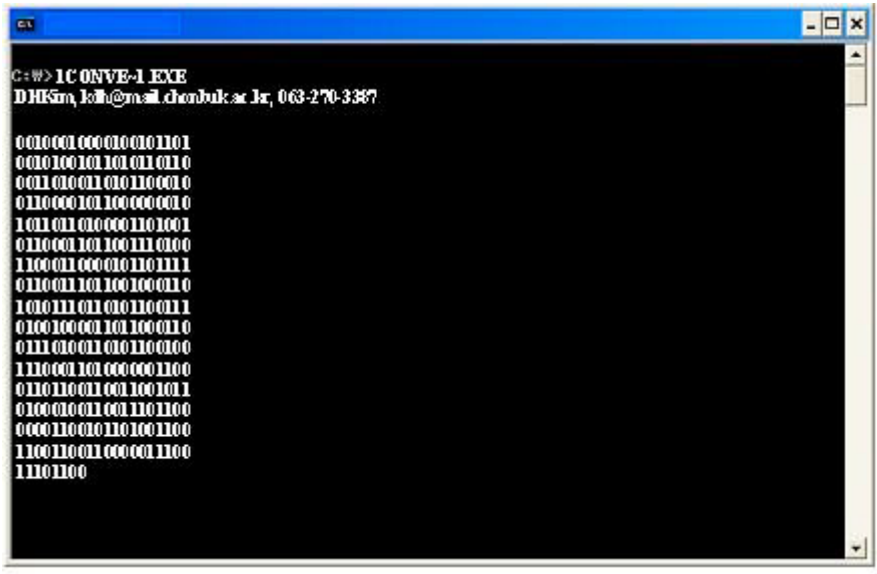


Fig. 2. Watermark Formation with information of copy owner

3.3 Algorithm for Inserting Watermark

Suggesting algorithm in the paper consists of 3 steps and the block diagram for these steps is like Fig. 3.

Step 1. DCT Conversion

By using contour detection from an original image, only the part that is decided as a domain to insert watermark is divided with 8*8 block. Then image is converted to frequency domain through DCT conversion.

Step 2. Insertion of watermark

The converted DCT modulus gets DCT modulus by performing zig-zag scan like JPEG compression algorithm. Except for low and high frequency, watermark is inserted to the middle frequency.

Step 3. Image formation of watermark

Through performance of IDCT to the changed information of image, watermark treated image is produced.

3.4 Algorithm For Watermark Detection

Suggesting algorithm for watermark detection in the paper consists of 3 steps and the block diagram for these steps is like fig. 4.

Step 1. DCT conversion

By using contour detection from watermarking inserted image, watermark inserted domain is divided with 8*8 block. Then image is converted to frequency domain through DCT conversion.

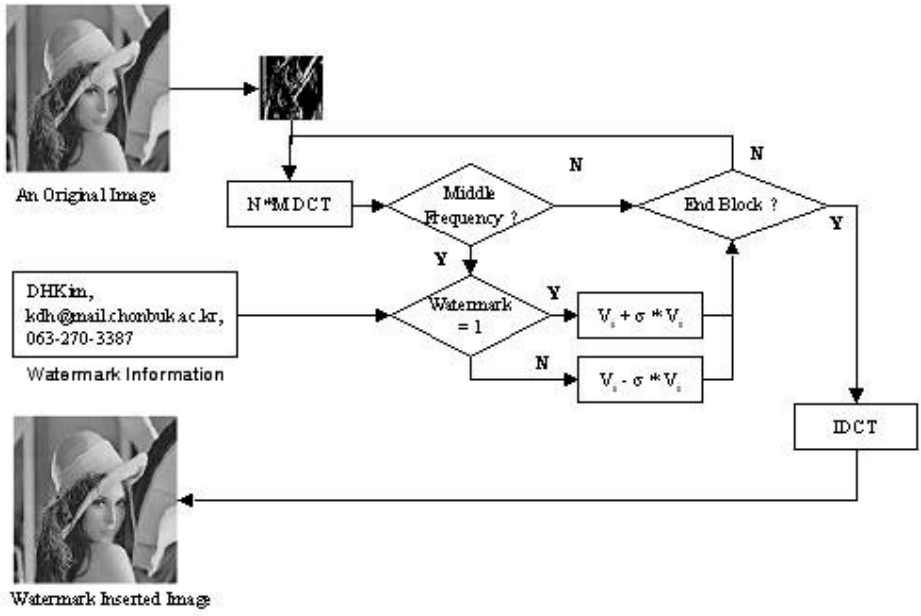


Fig. 3. A block diagram for Inserting Watermark

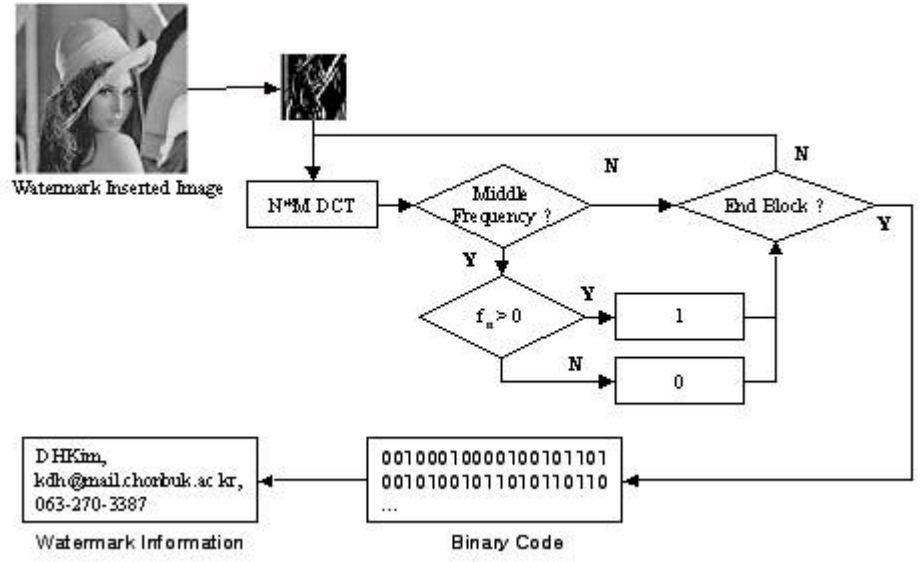


Fig. 4. A block diagram for Watermark Detection

Step 2. Detection of watermark

The converted DCT modulus gets DCT modulus by performing zig-zag scan. Except for low and high frequency, if element of the middle frequency is bigger than threshold then it gets 1 as watermark information and if element of the middle

Table. 1. Result of experiment for proposed algorithm

An Original Image	Contour Detection	Watermark Inserted Image

Table. 2 Result of experiment for image process techniques like JPEG

Result of experiment for image process techniques like Brightness, Contrast and Copy is Table 2. It's detected watermark all image process frequency is smaller than threshold then it gets 0 as watermark information and forms binary code.

Step 3. Confirmation of watermark

Convert assembly of binary code to information of copy owner. According to order, convert to 10 antialgarithm per 8bit and convert it again to ASCII code to show it as a

letter. If converted content indicates specific user, it is classified into watermark inserted image.
Result of experiment for proposed algorithm is Table1.

4 Conclusion

The rapid development of digital technology can be easily written, transmitted, distributed, and duplicated. When media provides data like image, sound, video, etc, everybody can have easy access to it. To speak this in other words, a person who got information illegally may tell people that he is the copy owner.
Therefore, it is very critical to protect one's intellectual property right from actual copy owner of digital information. Digital watermarking is an excellent way to resolve such problem.

In this paper, it proposes digital watermarking algorithm that can protect copy right of an image by using contour extract techniques. In an image, the area where has number of contour has very complicated images so can insert much amount of watermark, which results satisfaction of firmness and non-visibility of watermark.
In addition, the information of watermark converts a copy owner's information to ASCII code and by using converted ASCII code to binary code of 8 bytes, it does not concern similarity between watermarks in case of their distinction. Also, it suggests watermarking algorithm that does not require an original image and information of an inserted watermark.

Thus, digital watermarking algorithm by using contour detection in the thesis is a very good tool to protect copy right without using an original image and watermark information when there is illegal use of digital image. In addition, it is possible to insert watermark to an original image as it maintains non-visibility even though there has been numbers of watermark insertions in order to keep firmness of watermark

References

1. Mauro Barni, Franco Bartolini, Vito Cappellini, Alessandro Piva, "Copyright protection of digital images by embedded unperceivable marks", Image and Vision Computing 16, 1996.
2. W. Bender, D. Gruhl, N.Morimoto, A. Lu, "Techniques for data hiding", IBM Systems Journal, 1996.
3. J. Cox, J. Kilian, F.T.Leighton and T Shammoon, "Secure Spread spectrum watermarking for multimedia", IEEE Transactions on Image Processing, 1996.
4. Z. M. Lu and S.H, "Digital image watermarking technique based on vector quantisation, Electronic Letters, Vol 36 Issue4, 2000
5. Sungsoonthorn, Ratchata, "A Watermarking Technique Based On Wavelet Packet Transform", SCI 2001/ISAS2001, 2001.
6. I. Pitas, "A Method for Signature Casting on Digital Image", IEEE Int, Conf. On Image Processing. Vol 3, pp. 219-222, 1996.
7. S. Mallat, "A Wavelet tour of Signal Processing", Academic Press, 1998
8. G.C. Langelaar J.C.A vander Lubbe, J. Biemond, "Copy protection for multimedia data based on abling techniques", On Information Theory in the Benelux, 1996.

New Packetization Method for Error Resilient Video Communications

Kook-yeol Yoo*

School of Electrical Engineering and Computer Science, Yeungnam University, 214-1 Dae-Dong, Kyungsan City, Kyungpook 712-749 , South Korea,
kyoo@yu.ac.kr

Abstract. In this paper, we present an efficient video packetization method and associated video compression / decompression methods for the increase of error resilience of video communications over error-prone packet-switched networks. The main idea is inspired from the Multiple-Description Coding (MDC) strategy in which each image is separated into two sub-images like interlaced scanned image. The main contribution of this paper is the modification of RFC2429 packetization method and video compression and error concealment methods for the sub-images. The simulation results show that the proposed method significantly improves the subjective and objective quality compared with conventional RFC2429 packetization and TCON-based decoding method.

1 Introduction

With the broad deployment of Internet, the industrial demand on the video communication over the network is highlighted nowadays. The Internet video communication is fairly different from the traditional transmission over ISDN (Integrated Service Digital Network), in which QoS (Quality of Service) is guaranteed. For the Internet, which was originally developed for the data traffic, the network bandwidth has time-varying transmission delay and the packet can be lost due to network congestion [1, 2]. Since the video signal has the time-sensitive feature, the delay over the certain threshold, e.g., 150 msec, can be effectively transmission loss. The Internet video communication is complicated by the inconsistency between video signal and non-delay sensitive data-oriented Internet.

Most video compression standards have used the temporal DPCM (Difference Pulse Coded Modulation) techniques, called MC (Motion Compensation), which reveals excellent compression performance. The video bitstream coded with MC technique can cause the great loss of visual quality when the part of images is lost in the network. The error in a certain image can be propagated into subsequent images due to temporal dependency, whether those images are correctly received or not. For

* This work was supported by Yeungnam University Research Grant (105602).

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 329–337, 2004.
© Springer-Verlag Berlin Heidelberg 2004

the remedy such temporal error propagation, the Multiple Description Coding (MDC) methods are investigated [3]. The basic idea of the MDC is to make multiple descriptions for a single source and to encode those descriptions independently. So if one of the descriptions of the single image is impaired in the network, the other description is used to restore the lost description. Each description contains the independent information of the image and also the some information of other description, in other words, each description for the single images has correlation. The controlling the correlation is important issue in the MDC method.

Though the efficient description methods in MDC are investigated broadly, the associated packetization and restoration method at the decoder side are in early phase. The basic idea for configuration of the multiple descriptions is inspired from [4, 5], in which the image is up-sampled in vertical direction and then the up-sampled image are sub-sampled into two images. And the two images, in other words descriptions are encoded independently, requiring two video encoders at the transmission side and two decoders at the receiving side, i.e., double processing power is necessary. In this paper, we take more simplified approaches. The original image is reorganized into two descriptions as shown in Fig. 1.

(a) (b)

Fig. 1. Image reorangization method; (a) original image, (b) reorganized image

With the reorganized image, the video compression and packetization methods are proposed in this paper. Also the error recovery method when one of the subimages is losted, is proposed.

This paper is outlined as follows: In Section 2, the conventional error concealment methods and packetization methods are described and their relationship is investigated. The proposed method is explained in Section 3. The simulation results and conclusions are stated in Sections 4 and 5, respectively.

2 Conventional Packetization Method

2.1 Typical Video Communication Systems

A typical video communications system is depicted in Fig. 2. The original video source is first compressed by a video codec, and the compressed bitstream is then segmented into fixed or variable length packets and multiplexed with other data

types. The packets are sent over the network, possibly after channel coding. For the transport for the video over IP network, the UDP (User Datagram Protocol) is preferred protocol to rule out the re-transmission mechanism of TCP (Transmission Control Protocol). The upper layer protocol, RTP (Real-time Transport Protocol) is additionally used for the information on the packet losses, i.e., RTP sequence number [6]. Though these protocols for Internet video can be used for the identification of the lost packets, they do not provide any method to recover the quality of the damaged picture area. Video source decoder uses the correlation of damaged area with spatio-temporally surrounding received area to recover the damaged one. This recovery mechanism is called error concealment technique. Since the error concealment does not require any interaction with encoder side, it is suitable to video streaming application, i.e. 1-to-*N* type multicast transmission.

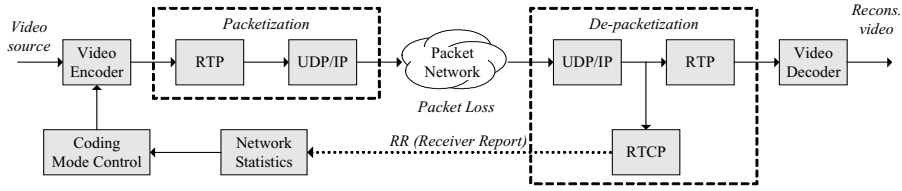


Fig. 2. A typical video communication system

2.2 RFC2429-Based Packetization Method

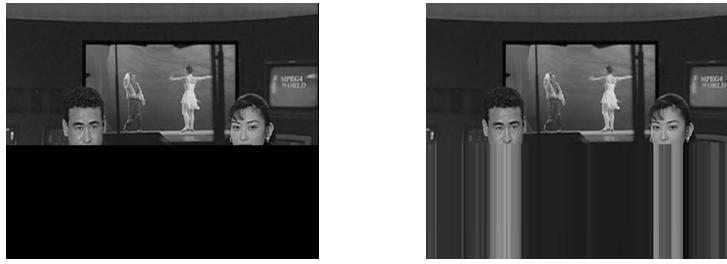
For the H.263 coded bitstream, the minimum unit of resynchronization unit is GOB layer. So the packetization is often performed in the unit of GOB. For instance, if a picture is packed into two packets, the content of each packet can be depicted as Fig. 3. If one of two packets is lost during the transmission over the network, then the decoder only receives the other packet. For instance, if the first packet is lost in the network, the result of decoding the image can be depicted as Fig. 4-(a). The error-concealed image by using spatial exterapolation is represented in Fig. 4-(b).

Packet header	GOB 0	GOB 1	…	GOB 7	GOB 8

Packet header	GOB 9	GOB 10	…	GOB 16	GOB 17

Fig. 3. Sequential packetization of a picture into two packets

On the other hand, the RFC2429 packetization method suggests the interleaved packetization as shown in Fig. 5 [7]. For this case, the damaged area of image in the lost packet can be recovered by using the interpolation of bottom and top pixels of upper and lower GOB's, respectively, as shown in Fig. 5 Compared with the sequential packetization method, the RFC2429 packetization method is substantially better reconstruction quality. However, the interpolation between 16 pixels produces poor visual quality as shown in Fig. 5. This low visual quality can be easily explained by



(a) (b)

Fig. 4. Spatial error recovery for the sequential packetization; (a) decoded image with the first packet, (b) spatially error concealed image

Packet header	GOB 0	GOB 2	…	GOB 14	GOB 16

Packet header	GOB 1	GOB 3	…	GOB 15	GOB 17

Fig. 5. RFC2429-based packetization

the correlation between pixels in 16-pixel distance, 0.482 as shown in Fig. 6. The Fig 7 shows that the pixels in one pixel distance has the correlation of 0.904, for two pixels distance, 0.822, and for three pixels distance, 0.768. So the spatial correlation above two pixels distance is very low to have good spatial interpolation performance. For the better quality recovery, we will propose the new packetization method in Section III, in which the error concealment for the packet loss recovery is achieved by the spatial interpolation of pixels in one pixel distance.

(a) (b)

Fig. 6. Spatial error recovery for RFC2429-based packetization; (a) decoded image with the first packet, (b) spatially error concealed image

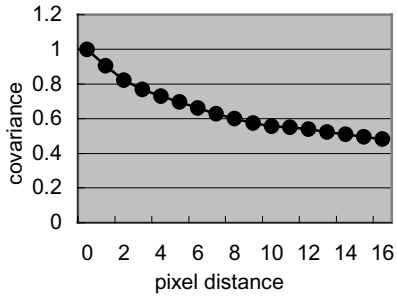


Fig. 7. Spatial correlation for the news sequence in vertical direction

3 Proposed Video Coding and Associated Packetization Methods

The main goal in the proposed packetization method is to improve the error concealment performance at the decoder side. As investigated in the Section II-C, the performance of the spatial error concealment is determined by the pixel distance between lost pixels and correctly decoded pixels. Though the RFC2429 based packetization method provides closer pixel distance than the sequential packetization method, the pixel distance for spatial interpolation is still too large. To reduce the pixel distances contained in the different packets, we reorganized the picture as shown in Figs. 1 and 8.

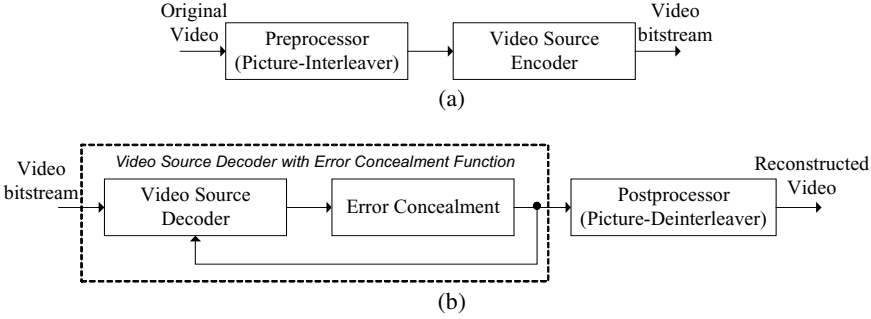


Fig. 8. Error resilient video coded with interleaved input source; (a) Video source encoder with interleaved input source, (b) Video source decoder with error concealment with interleaved input source

3.1 Proposed Video Coding Methods

For the simplicity of explanation, we first define the macroblock coordinate. Let us define $m_1(i, j)$ as the macroblock at (i, j) position of macroblock coordinate. The macroblock coordinate (i, j) for the luminance and chrominance signals with the 4:2:0 color format correspond to the pixel position (16i, 16j) and (8i, 8j), respectively. Let us further define $m_2(i, j)$ as the macroblock at (i, j) position of the modified mac-

roblock coordinate. The modified macroblock coordinate (i, j) for the luminance and chrominance signals with the 4:2:0 color format correspond to the pixel position $(16i, 16j + H/2)$ and $(8i, 8j + H/4)$, respectively, where the H represents the height of the luminance image. So $m_1(i, j)$ and $m_2(i, j)$ are located at the same positions in the respective sub-images. For instance, $m_1(0, 0)$ represents the uppermost and leftmost 16x16 luminance and 8x8 chrominance blocks in the upper partition, while $m_1(0, 0)$ represents the uppermost and leftmost 16x16 luminance and 8x8 chrominance blocks in the lower partition.

Proposed video coding method 1 (PVC1): For the compression of image, the upper partition and lower partition are compressed independently. So the motion estimation and coding mode decisions for $m_1(i, j)$ and $m_2(i, j)$ are independently. And also the boundary of the partitions is regarded as picture boundary. For instance, the motion vector prediction of the macroblock in the first GOB of the second partition cannot use the motion vectors belonging to the first the partition. For this independency and standard compliancy, the first GOB in the second partition should be encoded with GOB header.

Proposed video coding method 2 (PVC2): The coding parameters of the macroblocks, $m_1(i, j)$ and $m_2(i, j)$ are calculated jointly. For instance, both $m_1(i, j)$ and $m_2(i, j)$ are used as the current block in the motion estimation. But the encoding itself is conducted independently for standard compliancy, i.e., motion vectors of the $m_1(i, j)$ and $m_2(i, j)$ are independently encoded.

3.2 Proposed Packetization Method

If the RFC2429 based packetization method is used for the proposed algorithm and two packets per image is formed, the first packet contains the image contents from both partitions. So if one of the two packets is lost, the description of some part of image is not available. To preserve the independency of each partition in transmission, the sequential packetization is used in the proposed system as shown in Fig. 3. Each partition contains the picture header information, i.e., for the second partition, the redundant picture header is inserted. This redundant picture information is a part of RFC2429 method. So the proposed packetization method can be regarded as a modified RFC2429 method.

3.3 Proposed Video Decoding Method

The partitioned image in the lost packet is restored by using the other partition in the correctly received packet in the same picture. This kind of restoration is called error concealment technique. The error concealment method is divided into two categories: the spatial error concealment and the temporal error concealment. In the spatial error concealment, the lost pixel value is restored by interpolating the correctly received neighboring pixels. On the other hand, the lost pixel is reproduced based on the mo-

tion compensation by using previous image and the motion vectors used in the motion compensation is derived from the motion vectors in the correctly received neighboring macroblocks.

Since the pixel distance between lost and correctly received pixels is just one pixel in the proposed video coding and packetization method, the spatial interpolation can result in good restoration. Also temporal error concealment can be used in the proposed system. Since the motion vectors from two partitions have strong correlation in PVC1 or are exactly the same in PVC2, the temporal error concealment can be effectively conducted.

4 Simulation Results

The simulation environment used in this paper is compliant to the common test condition [7, 8] used in the standardization of ITU-T H.263 [9]. The test sequence used in the simulation is composed of the 100 frames of 'News' sequence in CIF (Common Intermediate Format; 352 pixels per line and 288 lines for luminance and the horizontally and vertically 2:1 sub-sampled version for chrominance, i.e., 4:2:0 format) format and temporal resolution is 10Hz (10 frames per second). The Internet packet loss rates used in the simulation are 3%, 5%, and 10% [7]. The basic transmission protocols are the RFC2429/RTP/UDP/IP and modified RFC2429/RTP/UDP/IP for the conventional and proposed methods, respectively. The video compression method for the conventional method is the baseline codec of ITU-T H.263, i.e., H.263 standard without any Annex coding tools and the proposed methods use the ITU-T H.263 baseline with modifications described in the Section 3.1. The video coding bitrate is fixed to 128 kbps. The cost function used in the performance comparison is PSNR (Peak Signal to Noise Ratio). The comparing methods are summarized in Table 1.

Table 1. Summary of the comparing methods

Method	Video codec	Packetization	Error concealment
CONV	H.263	RFC2429	TCON [10]
PROP1	PVC1	Modified RFC2429	Spatial concealment
PROP2	PVC2	Modified RFC2429	Spatial concealment

The results for each frame is depicted in Fig. 9. Since the proposed coding method converts image into two partitions as shown in Fig. 1, the vertical correlation in the input images are reduced, i.e., the pixel distance in the Fig. 1-(b) is effectively 2 pixels, while the conventional method is 1 pixel. So the first frame and subsequent frames show low coding efficiency, compared with conventional method (See 'no error case' in Fig. 9). When there, however, is the channel error, i.e., 5% loss case in Fig. 9, the proposed method improves significantly over the conventional method due to better error concealment at the decoder side. The average PSNR is summarized in Table 2, showing the effectiveness of the proposed methods. Comparing the proposed methods, PVC1 and PVC2, since PVC1 uses best motion and coding parameters for

each macroblock, the PVC1 gives better coding efficiency than PVC2, as shown in the 'No packet loss' cases of Fig. 9. However, the motion vectors in PVC2 are the same for both partitions, in other words, the motion vector in PVC2 is estimated considering the both macroblocks from the upper and lower partition. So if the temporal error concealment is used at the decoder side, the performance of the PVC2 can be improved. In this paper, this extension to the temporal error concealment, which requires selection method between spatial and temporal error concealment for each lost macroblock at the decoder side, is leaved as our further study.

Table 2. Performance comparison in PSNR [dB] with respect to the packet loss rate

Method	No packet loss	3%	5%	10%	Average (loss)
CONV	35.45	28.64	26.94	25.55	27.04
PVC1	31.94	29.05	28.41	25.87	27.78
PVC2	31.11	28.61	27.93	25.68	27.41

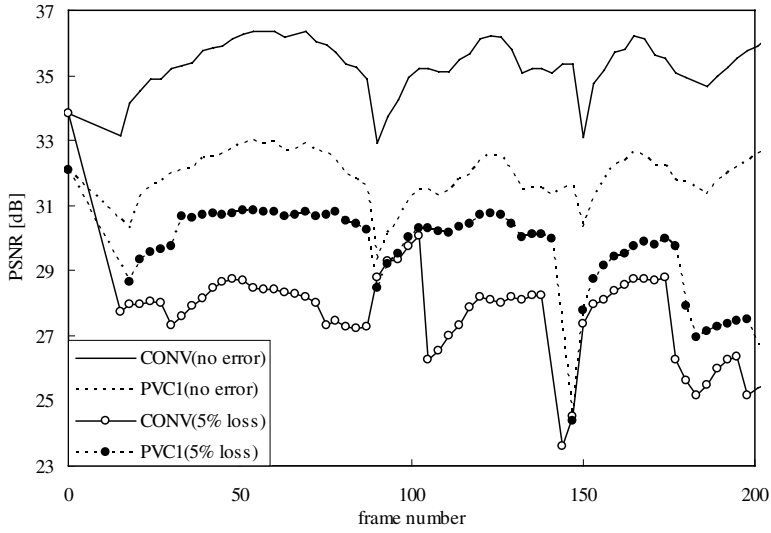


Fig. 9. PSNR comparison for each frame for the error free and 5% packet loss cases

5 Conclusion

In this paper, we present an efficient video packetization method for increasing the error-resilience of video communications over error-prone packet-switched networks. The quality of Internet video is determined by the two facts. Firstly, the video data transmitted over the packet network are lost in packet. Second, the lost portion of image is restored by the error concealment by using the uncorrupted part of the im-

age. The packetization is, therefore, highly coupled with error concealment. In the proposed packetization method, the picture is re-organized into two sub-images like two fields of a frame in the interlaced video. Each sub-image is independently encoded by using conventional video codec and the encoded sub-image is, then, packetized into separately packets. There, therefore, exists a pair of packets in a picture, whose contents in a picture has strong spatial correlation. The lost packet can be easily recovered by using the data from the other packet in the pair. For the better error resilience, the coding mode decisions for the paired packets are also presented in this paper. The proposed method will be strong tool to guarantee the QoS for the video communication over error prone network. As a further work, we investigate the selection method among the spatial and temporal error concealment at the decoder side.

References

1. B. A. Forouzan: TCP/IP protocol suite: McGraw Hill (2000)
2. J. Davidson and J. Peters: Voice over IP fundamentals, Cisco press (2000)
3. A. El Gamal and T. Cover: Achieval rates for multiple descriptions, IEEE Trans. Information Tehory, vol. 28 (1982), 851-857
4. Shahram Shirani, Michael Gallant, and Faouzi Kossentini: Multiple description coding using pre- and post-processing, in Proc. of IEEE Information technol. Coding and computation, (2001) 35-39
5. Michael Gallant, Shahram Shirani, and Faouzi Kossentini: Standard-compliant multiple description video coding, in Proc. of IEEE ICIP vol. 1 (2001), 946-949
6. H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson: RTP: A transport protocol for real-time applications," IETF RFC1889, (1996) [Online] Available from http://www.ietf.org
7. Stephan Wenger: Proposed error patterns for Internet experiments, ITU-T VCEG Doc. # Q15-I-16 (1999)
8. Max Luttrell, Stephan Wenger, and Michael Gallant: New versions of packet loss environment and pseudomux tools, ITU-T VCEG Doc. # Q15-I-09 (1999)
9. ITU-T Recommemdation H.263, Video coding for low bitrate communications, ITU-T (1995)
10. Thomas R. Gardos: Video codec Test Model, Near-Term, Version 10 (TMN10) Draft 1, ITU-T VCEG Doc. # Q15-D-65d1 (1998)

A Video Mosaicking Technique with Self Scene Segmentation for Video Indexing

Yoon-Hee Choi, Yeong Kyeong Seong, Joo-Young Kim, and Tae-Sun Choi

Dept. of Mechatronics, K-JIST
1 Oryong-dong, Buk-gu, Gwangju, 500-712, South Korea.
dals@sipl.kjist.ac.kr

Abstract. It is important to extract key information from video for the purpose of indexing and fast scene retrieval. Conventional frame-based video representation is appropriate for viewing it in a movie mode, but is not adequate for efficient access to information of interest. Therefore, Scene-based video representation using an image mosaicking for video indexing has been proposed recently. The scene segmentation is the first step of an image mosaicking because a mosaic image is composed of background of all frames that comprise the scene. Therefore, the image mosaicking with simultaneous scene segmentation is the natural choice for an efficient video representation. In this paper, we present an image mosaicking algorithm with efficient and robust automatic scene segmentation using phase correlation and motion-based algorithm. Simulation results show that the proposed method is fast and robust for the scene change detection and appropriate for scene-based video indexing.

1 Introduction

As the amount of multimedia data is increasing, browsing of videos in the multimedia database becomes important. Conventional frame-based video representation is appropriate for viewing it in a movie mode, but is not adequate for efficient access to information of interest. Therefore, scene-based video representation using an image mosaicking for video indexing has been proposed recently [1, 4]. This representation can be used as a video summary to browse the contents of the whole sequences and quickly access the shots of interest and play from that scene. In addition to video indexing applications, a mosaic representation of images has been applied to various areas such as panoramic view [2] and video compression [3].

Shot boundary detection is indispensable for the image mosaicking as one video sequence is composed of many different subsequences. The temporal video segmentation is very important in video analysis because most of the scene analysis starts from it. For this reason, many shot boundary detection algorithms also have been proposed in last decades.

Conventionally, video segmentation and image mosaicking have been developed independently. The separate works for image mosaicking and video segmentation are

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 338–347, 2004.
© Springer-Verlag Berlin Heidelberg 2004

not an efficient way if there exist some novel methods for image mosaicking with simultaneous video segmentation. In this paper, an image mosaicking algorithm is proposed with efficient and robust automatic scene segmentation. Phase correlation and motion compensation are used for image mosaicking and scene segmentation.
The rest of the paper is organized as follows. In section 2, previous works for scene segmentation and video mosaicking algorithms are described. Section 3 gives a detailed description of global motion estimation using phase correlation and computation of mosaic images. A proposed cut detection algorithm for scene segmentation is described in section 4. In section 5, simulation results are presented to show the performance of the proposed algorithm. Conclusions are given in section 6.

2 Previous Works

2.1 Scene Segmentation Algorithms

Considerably many algorithms have been reported on detecting shot boundaries. Most of the shot boundary detection algorithms can be classified into spatial domain based methods, MPEG compressed domain based methods and temporal slice based methods. The simplest way for detection is to measure difference between successive frames [7]. Frame difference larger than some threshold gives shot boundary point. However, this method is very sensitive to the fast object or camera motion. To overcome this drawback, several motion-based algorithms have been proposed [13]. However motion-based algorithms also suffer from high computational cost for block matching. The histogram-based approach can relax the problem of sensitivity to fast object and camera motion. The histogram-based methods find the difference between histograms of consecutive frames and large frame differences are considered as possible shot changes. Gargi *et al*. experimented with various techniques to compare the performance of each algorithm [13]. They concluded that color histogram based methods are the most robust methods with a moderate computational cost. However histogram based methods are sensitive to lighting and global gray level changes.
Temporal slice based methods transform a 3-dimensional video sequence into a 2-dimensional temporal abstract image. M. G. Chung *et al*. [9] called this temporal slice as visual rhythm. They sub-sample a group of pixels from a frame and construct a visual rhythm. Then, they find vertical edge using horizontal differentiation from the visual rhythm and calculate vertical projection. They use statistical data of this projection to determine shot boundary. C. W. Ngo *et al*. [10] use a model that extracts the color-texture features from slices and captures the shape and orientation of regional boundary as model energy. They find shot boundary using the function of this energy. The performance of these methods is good, however they require scanning of all frames to construct visual rhythm.
Most of MPEG compressed domain-based methods utilize DCT coefficients. B. L. Yeo and B. Liu use DC images extracted from the compressed video [15]. This method is based on some measures derived from histograms of DC images. However

it needs much computational cost to get DC images from P and B frames. Gargi *et al.* simulated six different MPEG compressed domain based algorithms and conclude that MPEG compressed domain based methods are fast, run in real time, but do not perform as well as the color histogram based methods [13].

2.2 Video Mosaicking Algorithms

The image mosaicking technique is roughly classified into the intensity-based methods, feature-based method and frequency-domain based methods.
The intensity-based methods usually use Levenberg-Marquardt algorithm to optimize the registration parameters. R. Szeliski [2] uses hierarchical matching or phase correlation technique to estimate initial global motion and Levenberg-Marquardt algorithm for local image registration.
The feature-based methods use feature matching between two images that have to be registered. To find exact feature correspondence is the most important problem in these methods. [11] and [12] proposed contour-based approach to extract features and find the registration parameters. They use LoG (Laplacian-of-Gaussian) operator to find the contours. The modified chain code matching is used to reduce the matching contour candidates. Shape attributes are computed to find the matching contours. A consistency check is performed to eliminate the false matching after contour matching process.
The frequency based methods use FFT (Fast Fourier Transform) to find the registration parameters. Q. S. Chen *et al.* [5] calculate Fourier-Mellin invariant descriptor for each image and match the Fourier-Mellin invariant descriptors. In this method all parameters (translation, rotation, scale) can be calculated using phase correlation method because all parameters can be modified into translational form.
The image mosaicking technique has been applied to extend the field of view of satellite images. However, the area of image mosaicking application includes the video indexing as development of digital video technology. Michal Irani *et al.* [1, 4] have introduced the concept of video indexing using a mosaic image. They classified the mosaic into static and dynamic mosaic, and presented various application of mosaic representation.

3 Global Motion Estimation and Mosaic Image

3.1 Global Motion Estimation by Phase Correlation

The phase correlation is based on the Fourier shift theorem, *i.e.*, a shift in the spatial domain is equivalent to a phase shift in the frequency domain.
Let image $f_2(x,y)$ be a shifted replica of image $f_1(x,y)$ displaced by (x_0, y_0) *i.e.*,

$$f_2(x,y) = f_1(x - x_0, y - y_0) \tag{1}$$

The Fourier transform on both sides of equation (1) gives:

$$F_2(\omega_x,\omega_y) = F_1(\omega_x,\omega_y)e^{-j(\omega_x x_0+\omega_y y_0)}, \tag{2}$$

where F_1, F_2 indicate the Fourier transform of f_1 and f_2 respectively. Multiplying $F^*_2(\omega_x,\omega_y)$ on both sides of equation (2),

$$\left|F_2(\omega_x,\omega_y)\right|^2 = F_1(\omega_x,\omega_y)F_2^*(\omega_x,\omega_y)e^{-j(\omega_x x_0+\omega_y y_0)}, \tag{3}$$

where, F^* is the complex conjugate of F.

The cross-power spectrum of two images $f_1(x,y)$, $f_2(x,y)$ is defined as a normalized form of equation (3), *i.e.*,

$$\frac{F_1(\omega_x,\omega_y)F_2^*(\omega_x,\omega_y)}{\left|F_1(\omega_x,\omega_y)F_2(\omega_x,\omega_y)\right|} = e^{j(\omega_x x_0+\omega_y y_0)}. \tag{4}$$

It is obvious that the inverse Fourier transform of equation (4) gives displacement vector (x_0, y_0) of spatial domain as indicated in equation (5).

$$\begin{aligned}\Im^{-1}\left[\frac{F_1(\omega_x,\omega_y)F_2^*(\omega_x,\omega_y)}{\left|F_1(\omega_x,\omega_y)F_2(\omega_x,\omega_y)\right|}\right] &= \Im^{-1}\left[e^{j(\omega_x x_0+\omega_y y_0)}\right] \\ &= \delta(x_0,y_0)\end{aligned} \tag{5}$$

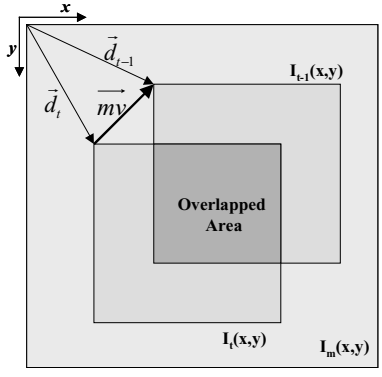


Fig. 1. Mosaic image construction

3.2 Computation of the Mosaic Images

The mosaic image construction procedure is depicted in Figure 1. If the image I_{t-1} is located at $\vec{d}_{t-1}$ and global motion vector is $\overrightarrow{mv}$, $\vec{d}_t$ and $\vec{d}_{t-1}$ are related as:

$$\vec{d}_t = \vec{d}_{t-1} - \overrightarrow{mv}. \tag{6}$$

In case of static mosaic [4], all frame locations, which compose one scene, must be identified on the mosaic coordinates to construct static mosaic image at a time. When one or both of the components of $\vec{d}_t = (d_{tx}, d_{ty})$ from equation (6) is negative, all

$\vec{d}_{t-1}, \vec{d}_{t-2}, \cdots, \vec{d}_0$ must be recalculated, because the origin of mosaic plane is changed. The updated $\vec{d}_{t-1}, \vec{d}_{t-2}, \cdots, \vec{d}_0$ are

$$\vec{d}_{new_l} = \vec{d}_l - neg(\vec{d}_t) \text{, where } l = t, t\text{-}1, t\text{-}2, \ldots, 0, \tag{7}$$

and

$$neg(\vec{v}) = \begin{cases} (0,0), & v_x \geq 0, v_y \geq 0 \\ (v_x,0), & v_x < 0, v_y \geq 0 \\ (0,v_y), & v_x \geq 0, v_y < 0 \\ (v_x,v_y), & v_x < 0, v_y < 0 \end{cases}.$$

The total mosaic image size at time t is

$$\begin{aligned} MW_t &= d_{(t-1)x} - mv_x + MW_{t-1} \\ MH_t &= d_{(t-1)y} - mv_y + MH_{t-1} \end{aligned}, \tag{8}$$

where MW_t and MH_t are mosaic image width and height at time t respectively.
For the dynamic mosaicking [4], a previous mosaic image must be aligned to the current mosaic image. The previous mosaic image position on the current mosaic plane must be defined.

$$\begin{aligned} dm_x &= \begin{cases} mv_x - d_{(t-1)x}, & d_{(t-1)x} < mv_x \\ 0, & \text{otherwise} \end{cases} \\ dm_y &= \begin{cases} mv_y - d_{(t-1)y}, & d_{(t-1)y} < mv_y \\ 0, & \text{otherwise} \end{cases} \end{aligned}, \tag{9}$$

where, dm_x and dm_y are x, y coordinates of previous mosaic image in the current mosaic plane respectively.

4 Cut Detection Algorithms

When only the translational camera motion is assumed, a mosaic image can be constructed with simultaneous scene segmentation using phase correlation. In this case, if peak value of correlation surface is less than a certain threshold value σ_1, shot boundary is suspected. Final decision of shot boundary is declared according to following MSE (Mean Square Error) criteria.

$$\eta(t) = \frac{1}{A} \sum_{(x,y) \in A} \left[I_t(x,y) - I_{t-1}(x,y) \right]^2, \tag{10}$$

where, A denote the overlapped area in Figure 1. If $\eta(t)$ is larger than certain threshold value σ_2, shot boundary is declared between the video frames I_{t-1} and I_t.
If camera motion contains other than translation, shot boundary can be detected without image mosaicking using following process [9]. At first, following local statistics should be computed to use at the decision stage.

$$\mu(t) = \frac{1}{N(W)} \sum_{k \in W} \eta(t+k) \tag{11}$$

$$\sigma^2(t) = \frac{1}{N(W)-1} \sum_{k \in W} [\eta(t+k) - \mu(t)]^2 \tag{12}$$

whrere, $\mu(t)$ is the sample mean of $\eta(t)$ and $\sigma^2(t)$ is the sample variance of $\eta(t)$ with sample window W at frame t. Here, W does not include a sample at time t and $N(W)$ is the number of samples in the window W. The following adaptive thresholding scheme decides the shot boundary points. In Eq. (13), if $c(t)$ is one, we can declare that the shot boundary is occurred at frame t.

$$c(t) = \begin{cases} 1, & \eta(t) > \mu(t) + K\sigma(t) \\ 0, & \eta(t) \le \mu(t) + K\sigma(t) \end{cases} \tag{13}$$

It is known that the choice of K value is quite arbitrary, but values from 3 to 8 work quite robustly [9]. We use 3.5 for K value in this paper.

5 Simulation Results

For the simulation of proposed algorithm, two kinds of test video sequences were used. The first one is a 340-frames sequence, whose motion is translation only. This sequence was made by author, which is composed of 50 frames of coast guard sequence (stand test sequence with moving objects: 51 frame to 100), 150 frames of airplane sequence (taken by author) and 140 frames of scenery sequence (taken by author). The other video sequence is taken from the real relay broadcasting of soccer game (Korea-Japan FIFA World Cup 2002 - Italy vs. Ecuador). This second video sequence includes complex camera motion. Thus, a proper mosaic image cannot be achieved because the phase correlation model works well only under the translational motion sequence.

Figure 2 shows the selected frames (frame 51, 67, 83 and 100) of the first scene from the first video sequence. And Figure 3 shows the static mosaic image of this scene. It can be seen that the moving objects (two boats) are blurred out because the pixels on the moving object area are averaged with different intensity values as object moves. Figure 4 shows the dynamic mosaic image of same scene. Each mosaic image is updated with the most recent frame, thus, moving objects do not blur out in the dynamic mosaic image as seen in Figure 4. Figure 5 and 7 show the selected frames of the second and third scene from the first video sequence and Figure 6 and 8 are synopsis images of these two scenes represented by dynamic mosaic image. As can be seen from the Figure 4, 6 and 8, phase correlation is well adapted if test video has camera motion that is only composed of translation. Figure 9 shows the selected frames of the second scene from the second video sequence. The camera motion is composed of translation with other complex motion. Figure 10 shows the dynamic mosaic of this scene. A discrepancy can be seen in the middle area of the mosaic image. This is because camera motion contains other than translation. When we use phase correla-

tion, only the translational motion is assumed, thus the sequence that has motion other than translation can show a discrepancy in the mosaic image.

The first test video sequence has two abrupt scene change points. One is between frame 50 and 51, the other is between frame 200 and 201. Figure 11 shows the peak value distribution of phase correlation surface of each frame. It can be seen from the Figure 11 that frame 51 and 201 are the candidates for the possible cut points. It is obvious from the Figure 12 that frame 51 and 201 are shot boundary points. The threshold value used for σ_l was 0.25.

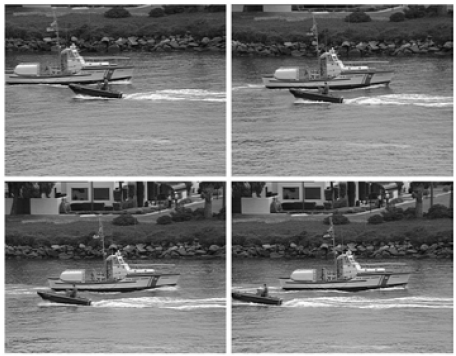


Fig. 2. Selected frames from the coast guard sequences

Fig. 3. Static mosaic image of coast guard sequence (51 to 100 frames)

Fig. 4. Dynamic mosaic image of coast guard sequence

The second test video sequence also has two abrupt scene change points. One is between frame 41 and 42, the other is between frame 137 and 138. Figure 13 and 14 show the peak value distribution of phase correlation surface and MSE distribution of the each frame. To find the scene change points, Eq. (13) was applied to MSE distribution of second video sequence. The open circle is MSE of overlapped area. The solid line is the mean value of W and dotted line is the STD (Standard Deviation) of W. The filled circle is the result of Eq (13) and represents shot boundary frames.

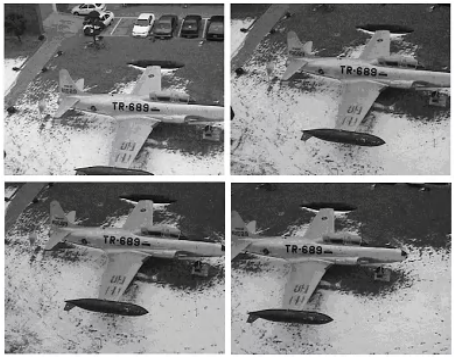


Fig. 5. Selected frames from the airplane sequences

Fig. 6. Dynamic mosaic image of airplane sequence

Fig. 7. Selected frames from the scenery sequences

Fig. 8. Dynamic mosaic image of scenery sequence

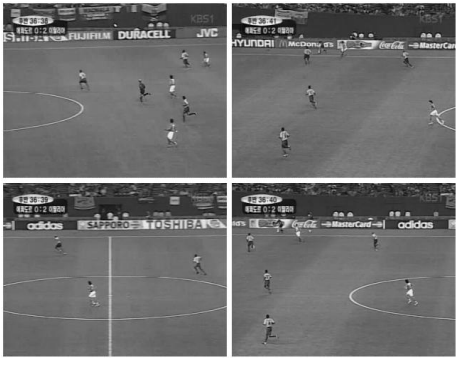


Fig. 9. Selected frames from the soccer sequence

Fig. 10. Dynamic mosaic image of soccer sequence

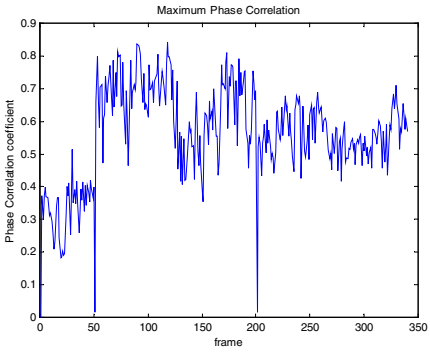


Fig. 11. Maximum values of phase correlation surface for the first video

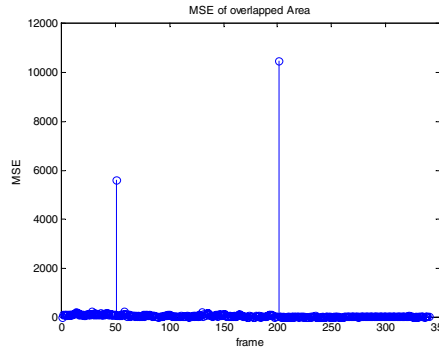


Fig. 12. MSE of overlapped area for the first video

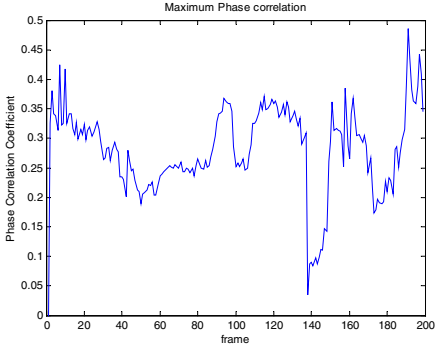


Fig. 13. Maximum values of phase correlation surface for the second video

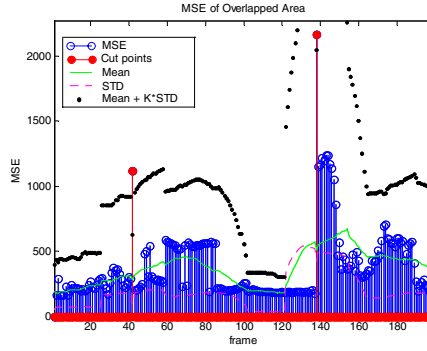


Fig. 14. MSE of overlapped area for the second video

6 Conclusions

In this paper, a new algorithm for an image mosaicking with automatic scene segmentation has been proposed. A phase correlation is usually used for image registration. Thus, in this paper, phase correlation was used for an image mosaicking and global motion-compensated cut detection algorithm was used along with phase correlation peaks. Simulation results showed that the proposed algorithm is fast and robust. In this paper, only translational motion was considered for an image mosaicking. However, phase correlation method can be easily extended to rotation and scale estimation [5, 6] and this work is in progress.

Acknowledgement. This work was supported by the Korea Research Foundation Grant (KRF-2003-041-D20470).

References

1. Michal Irani and P. Anandan, "Video indexing based on mosaic representations," Proceeding of the IEEE, vol. 86, no.5, pp. 905-921, May 1998.
2. Richard Szeliski, "Video mosaics for virtual environments," IEEE Computer Graphics and Applications, vol. 16, pp. 22-30, March 1996.
3. Michal Irani, Steve Hsu, P. Anandan, "Mosaic-based video compression," SPIE Proceedings, vol. 2419, February 1995.
4. Michal Irani, P.Anandan, and Steve Hsu, "Mosaic based representations of video sequences and their applications," Proc. of IEEE International Conference on Computer Vision, pp. 605-611, June 1995.
5. Qin-sheng Chen, Michel Defrise, and Deconinck, "Symmetric phase-only mached filtering of Fourier-Mellin transform for image registration and recognition," IEEE Trans. on PAMI, vol. 16, no. 12, pp. 1156-1168, December 1994.
6. B. Srinivasa Reddy and B.N. Chatterji, " An FFT-based technique for translation, rotation, and scale-invariant image registration," IEEE Trans. on Image Processing, vol. 5, no. 8, pp. 1266-1271, August 1996.
7. J. S. Borecsky and L. A. Rowe, "Comparison of video shot boundary detection technique," in Proceedings of SPIE, vol. 2670, pp. 170-179, 1996.
8. B. Shahrary, "Scene change detection and content-based sampling of video sequences," in Proc. SPIE/IS&T Symp. Electronic Imaging Science and Technology: Digital Video Compression, Algorithms and Technologies, vol. 2419, pp. 2-13, 1995.
9. M. G. Chung, H. Kim, and M. H. Song, "A scene boundary detection method," Proceedings of IEEE Iinternational Conference on Image Processing, pp. 933–936, 2000.
10. C. W. Ngo, T. C. Pong, and R. T. Chin, "Video partitioning by temporal slice coherency," IEEE Trans on Circuits and Systems for Video Technology, vol. 11, no. 8, pp. 941-953, August 2001.
11. H. Li, B. S. Manjunath, S. K. Mitra, "A contour-based approach to multisensor image registration," IEEE Trans on Image Processing, vol. 4, no. 3, pp.320-334, March 1995.
12. X. Dai, S. Khorram, "A feature-based image registration algorithm using improved chain-code representation combined with invariant moments," IEEE Trans on Geoscience and Remote Sensing, vol. 37, no. 5, September 1999.
13. U. Gargi, R. Kasturi, and S. H. Strayer, " Performance characterization of video-shot-change detection methods," IEEE Trans on Circuits and System for Video Technology, vol. 10, no. 1, February 2000.
14. A. Nagasaka and Y. tanaka, "Automatic video indexing and full-motion search for object appearances, " in Proc. IFIP 2nd Working Conf. Visual Database System, pp.113-127, 1992.
15. B. L. Yeo and B. Liu, "Rapid scene analysis on compressed video," IEEE Trans on Circuits and System for Video Technology, vol. 5, no. 6, December 1995.

Real-Time Video Watermarking for MPEG Streams

Kyung-Pyo Kang, Yoon-Hee Choi, and Tae-Sun Choi

Dept. of Mechatronics, K-JIST
1 Oryong-dong, Buk-gu, Gwangju, 500-712, South Korea.
tschoi@kjist.ac.kr

Abstract. In this paper, we propose a real-time video watermarking algorithm for MPEG streams. Watermarking Technique has been studied as a method to hide secret information into the signals so as to discourage unauthorized copy or attest the origin of the media. In the proposed algorithm, we take advantage of compression information of MPEG bitstreams to embed the watermark into I-, P-, and B-Picture. The experimental results show that the proposed watermarking technique results almost invisible difference between watermarked MPEG video and original MPEG video, and reduces the processing time. Moreover, it shows robustness against a variety of attacks as well.

1 Introduction

Nowadays, with the development of digital technology, it is easy to produce, edit and store digital images, audio and video, etc. From the situation, it's very difficult to distinguish original digital media (image, audio and video) from duplication. As comparing with the analog media, digital media copyright information can be more easily infringed by a pirate. This kind of problem makes digital copyright and protection of intellectual property go up to dispute. Digital Watermarks have been proposed to address this issue by embedding owner or distribution information directly into the digital media. The information is embedded by masking small modifications to the samples in the digital data. When the ownership of the media is in question, the information can be extracted to identify the owner or distributor.

In the literature, only a few compressed video watermarking schemes were proposed. F. Hartung et al. [2] proposed VWM algorithm based on the spread spectrum method. This method embeds the watermark into DCT coefficients of blocks of 8 by 8 pixels. Because the watermark is embedded into all frames, error caused by watermark embedding is propagated to the next frame to refer to the previous frame having the error. In order to block off the error, this scheme added a drift compensation signal. But it is the factor to increase the computational complexity. Langelaar et al. [4] proposed a VWM scheme preformed in the compressed domain. They added the label, which is considered as the watermark, to least significant bit (LSB) of selected variable length code (VLC) by changing another VLC that has the same run length, a level difference of 1 and the same codeword length. But, this algorithm is weak

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 348–358, 2004.
© Springer-Verlag Berlin Heidelberg 2004

against attacks. Recently, Langelaar et al. [5] proposed to embed the watermark after making energy differences of high frequency components by selectively discarding high frequency DCT coefficients in certain image regions. Error is not propagated because the watermark is inserted only into I pictures. But, blurring effect of edges regions can happen because AC coefficients of high frequency components of one region should be discarded to make constant energy difference between two regions.
In this paper, we deal with a novel compressed domain watermarking technique that is applicable to MPEG multiplexed streams (video and audio). We can get efficiency and robustness by using two different schemes during the embedding and detecting process. One is to embed the watermark into I picture using quantization parameter and quantized DCT coefficients, the other is to embed the watermark into P and B picture using variable length codes. Especially, the watermark can be embedded into raw video data in the middle of encoding and MPEG bitstreams as well. The embedding and detection is so fast that the proposed algorithm can be applied to a real-time video watermarking system. The resultant watermarked video sequences are shown to prove the robustness of our proposed algorithm.

2 Proposed Real-Time VWM Algorithm

2.1 Watermark Embedding

The watermark is embedded by two different schemes because the capacity to be embedded is different according to Picture Type. In I picture case, all macroblocks is intra-coded, and intra-coding is very similar to image compression such as JPEG compression. Therefore, image watermarking scheme can be applied to this type of picture. However, it is necessary to lower the computational complexity for a real time processing. So we embed the watermark into I Picture using the *relative complexity of a block* (RCOB) from quantized DCT coefficients and *QP* (quantization parameter) from Rate Control. Computing both factors does not require lots of a processing time because these factors are naturally computed through the encoding procedure. If such a scheme is applied to B and P picture, error propagation will be increased and this results in the quality degradation of video sequences. Therefore, real-time labeling scheme is applied to B and P picture [4]. In comparison with I picture, the amount of embedding watermarks is relatively small and this scheme is less robust that of I picture but, it has an advantage of the low computational complexity which makes real time processing possible.
Figure 1 shows the block diagram of watermark embedding in video encoder. Embedding the watermark in I picture is performed after quantization of DCT coefficients and embedding the watermark in P, B pictures is performed in the middle of variable length coding. The proposed watermark embedding is as follows

Step 1. Generate pseudo binary numbers with a secret key whose length is $L = 2 \times MB_cnt \times Block_Count$, where *Block_Count* is the number of blocks in a macroblock.

$$Pi = \{ P_k \mid P_k = \{0,1\}, k = 1 \ldots L \}$$

Step 2. Determine picture type after quantization. In this algorithm, there are two kinds of picture types. One is I picture, the other is B and P picture

[I Picture Case]

Step 3. RCOB is computed using equation (1). If RCOB is zero, the corresponding block should be skipped to prevent bit rates from increasing because embedding one bit in the block that has only DC component can increase bit rates. Otherwise, one bit should be embedded in a diagonal component.

$$RCOB = \sum_{i=0}^{63} \left| Q(DCT_{coeff}) \right|_i - \sum_{i=0,1,8,9} \left| Q(DCT_{coeff}) \right|_i \tag{1}$$

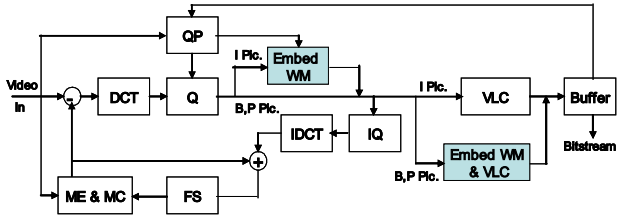


Fig. 1. Block Diagram of Watermark Embedding in Video Encoder.

RCOB is the absolute sum of quantized DCT coefficients which shows the relative activity of a block, but the greater RCOB does not exactly mean more active block because it can be greater if *QP* is small, even if the block does not have strong texture.

Step 4. Complexity of a block (COB) is computed using equation (2). If COB is greater than a threshold value, vertical factor (VF) and a horizontal factor (HF) is computed using equation (3) and (4), respectively, and one more bit should be embedded in a horizontal component (HC) or a vertical component (VC) in the corresponding block. If VF is greater than HF, extra one bit should be embedded in VC. Else if HF is greater than VF, extra one bit should be embedded in HC.

$$COB = QP \times RCOB \tag{2}$$

$$VF = \sum_{i=2,3,10,11} \left| Q(DCT_{coeff}) \right|_i \tag{3}$$

$$HF = \sum_{i=16,17,24,25} \left| Q(DCT_{coeff}) \right|_i \tag{4}$$

VF reflects the vertical edges and HF reflects the horizontal edges. By embedding extra bit, each edge component can be outstanding without the degradation of visual quality. During the embedding process, if the watermark is equal to zero, LSB of the corresponding quantized DCT coefficient and Zero are logically AND operated. Otherwise, the LSB and One are logically OR operated. Moreover, the reason of selecting the neighboring components of DC component for watermarking embedding is that these components have the robustness against the general attacks and have the strong possibility to have non-zero value compared to any other components. Especially, the

variation of diagonal edge is less sensitive to human visual system (HVS) compared to that of vertical or horizontal edge [6]. That is why a diagonal component is the main candidate and whenever a block has strong texture VC or HC can be the extra candidate for watermark embedding. Figure 2 shows the positions of a diagonal component, VC, and HC in a block and 4 candidates for VF and HF, respectively.

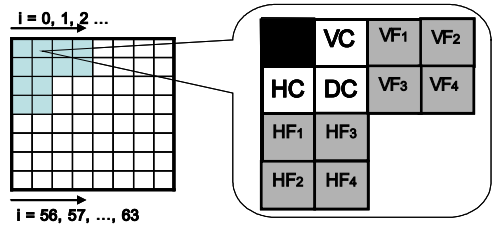


Fig. 2. Positions of each component and VF and HF in a block

[P/B Picture Case]
This scheme changes a variable length code (VLC) into another VLC which has the same run length, a level difference of 1 and the same codeword length. These VLCs are shown in Table 1.

Table 1. The Candidates of VLCs to be watermarked from Table B.14 of the MPEG-2 Standard.

Length	**9**	**13**	**14**	**15**	**12**	**17**
(Run, Level)	(0,5)	(0,8)	(0,12)	(0.16)	(1,10)	(1,15)
	(0,6)	(0,9)	(0,13)	(0,17)	(1,11)	(1,16)

In Table 1, if level is odd and watermark is Zero, or level is even and watermark is One, then its VLC should be replaced by the other. If level is even and watermark is Zero, or level is odd and watermark is One, its VLC should be not changed. Thus, LSB of the level represents the watermark bit.
Until now, the procedure of watermark embedding into raw video files in the middle of encoding has been described. Figure 3 shows block diagram of embedding the watermark into MPEG bitstreams. The embedding procedure is almost similar to the above explanation. Incoming MPEG bitstream is split into header and side information, motion vector and DCT encoded signal blocks, only the latter part of the bitstream is altered: motion vectors and head/side information remain untouched and are copied to the watermarked MPEG bitstream. However, the bit-rate should not be increased during the embedding process.
In order to reduce the increased bit-rate that is originated from watermark embedding into I picture, VLCs in B and P pictures are changed according to Table 2. So to speak, if (Run,Level) is (0,4), it will be changed into (0.3) and 2 bits will be reduced. This process will continue until the number of bits reduced approaches to that of bits increased in every GOP.

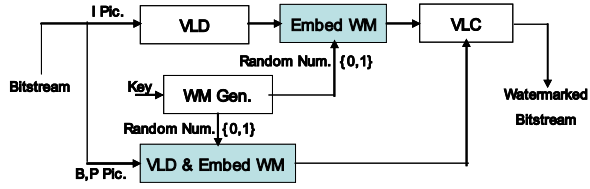


Fig. 3. Watermark Encoder for MPEG bitstreams.

Table 2. VLCs for reduction of bit-rate in Table B.14 of the MPEG-2 Standard .

VLC size	Run	Level	Diff. of Length
5+1	0	3	2
7+1	0	4	
8+1	3	2	4
12+1	3	3	

2.2 Watermark Detection

Watermark extraction is exactly reverse procedure of embedding procedure. If the type of a current frame is I type, RCOP, COB, VF, and HF are computed according to the condition of a block and the watermark is extracted. Then, LSB of a corresponding quantized DCT coefficient and One should be logically AND operated. After this operation of the whole blocks in the picture, watermark detection is performed with the resultant bit stream. In case that the current frame is P or B Picture, the extracted watermark is One if the level of VLC in Table 1 is odd,. Otherwise, it is Zero.
The resultant bit stream, $P_k^{'}$ *(k=1,2,...L)*, will be computed using equation (5) for the watermark detection.

$$DV = \sum_{k=1}^{L} \left\{ p_k \odot p_k^{'} - p_k \oplus p_k^{'} \right\} \tag{5}$$

In equation (5), DV is a detection value and DV is computed with $P_k^{'}$ and lots of P_ks which are generated by their corresponding seeds.

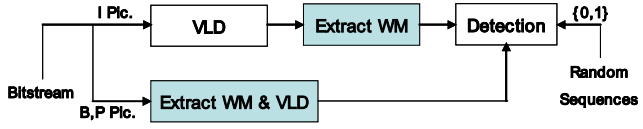


Fig. 4. Block Diagram of Watermark decoder.

3 Experimental Results

In order to confirm that the proposed real-time VWM is effective and robust, we performed some numerical experiments with four video data. First video sequence is

"flower garden" which is comprised of 150 frames and second video sequence is "football" which is comprised of 210 frames. Each luminance frame is of size 352X240. Third video sequence is "Miss America" which is comprised of 111 frames and fourth video sequence is "Salesman" which is comprised of 200 frames. Each luminance frame is of size 352X288. Frame rate, defined as the number of frames per second, is 25 PPS (Pictures per Second). The number of frames in GOP is 12 frames and the distance between I and P frame is 3 frames. The format used for chrominance is 4:2:0. The same threshold value, which was set at 200, was applied. Bit rates will be varied with 1.5Mbps, 1.2Mbps, and 0.6Mbps and Table.3 shows the average number of embedded bits in three types of picture according to different bit rates.

Table 3. Average number of embedded bits.

Bitrate	Pic.Type	flower	football	miss a	salesman
0.6 Mbps	I	1848.9	1296.4	1430.6	2165.6
	P	8.5	51.1	8.5	27.3
	B	0.3	8.5	0.2	0.9
1.2 Mbps	**I**	**2536.0**	**1960**	**2052**	**2637.9**
	P	**106.4**	**282.3**	**29.1**	**108.2**
	B	**5.8**	**62.2**	**1.2**	**7.3**
1.5 Mbps	I	2647.8	2105.7	2222.4	2751.2
	P	195.4	421.5	51.2	168.8
	B	12.9	106.0	2.2	13.2

From these results, the number of embedded bits depends on bit rate. That is to say, the higher the bit rate is, the more the number of embedded bits is. More watermark bits were embedded into I Picture because Flower has more complex blocks than Football. However, Football has lots of VLCs to which the watermark can be embedded compared to Flower because it has scene change and lots of motion blocks. On the other hand, the similarity between two successive frames of Flower is relatively high because Flower only has slow camera motion moving from left to right. This fact results in the small number of VLCs which the watermark can be added. If we compare Miss America with Salesman, neither of them has camera motion. But, Salesman is more complex because it has lots of edges and slightly more motion blocks than Miss America. In order to evaluate the quality of the watermarked video objectively, PSNR (Peak to Signal and Noise Ratio) of each frame was computed for two video sequences. The definition of PSNR is

$$PSNR = 10 \log_{10} \frac{255^2}{MSE} \tag{6}$$

,where MSE is Mean Square Error. Figure 5 shows PSNR of the original MPEG and watermarked video sequences. Here, the upper line indicates PSNR of the original MPEG video sequence and the lower line indicates PSNR of the watermarked video sequence. From Figure 5, the PSNR difference is very small and partially very similar.

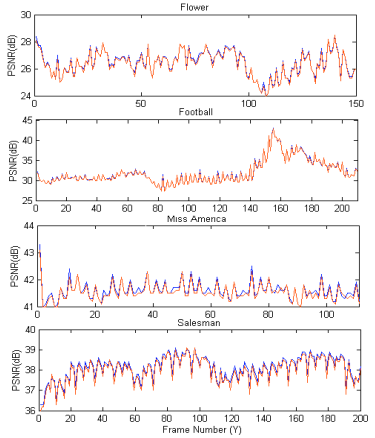


Fig. 5. PSNR of Y at 1.2 Mbit/s.

Fig. 6. (a) Flower garden (b) Football (c) Miss America (d) Salesman: original (left), watermarked (middle), and difference (right).

If the original I picture is subtracted from the corresponding watermarked I picture and the absolute value of difference signal is amplified by 10, the images, shown in Figure 6 (a) and (b), are obtained. In case of Figure 6 (c) and (d), the absolute value is amplified by 20. Four video sequences are coded at 1.2Mbit/s and differences are introduced because of watermark embedding. According to Figure 6 most differences are located around the edges and in the textured areas. The smooth areas are left unaffected. The watermark was extracted perfectly if a video stream didn't get an attack. In this experiment, we computed detection value of B Picture but didn't try to detect because embedded bits in B Picture is relatively so small that it is insignificant to compare the extracted bits with binary random numbers which is generated by any key number. Figure 7 shows the result of watermark detection experiments.

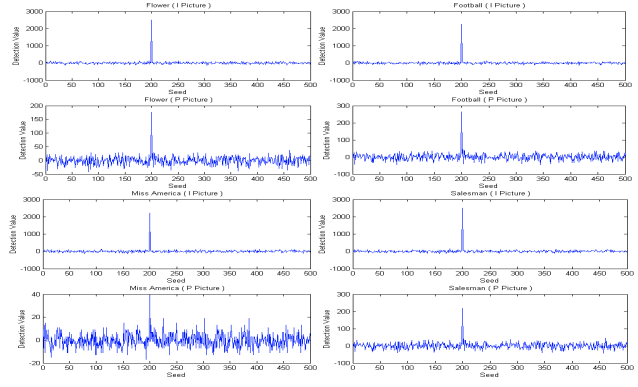


Fig. 7. Detection Value of I and P Picture.

We computed the detection values under various attacks including additive Gaussian noise (variance of 1), low-pass filtering (3×3), median filtering (3×3), histogram equalization, and MPEG re-encoding. The watermark was detected from a randomly selected I Picture. . Figure 8, 9, 10, 11 and 12 show the detection values after these attacks.

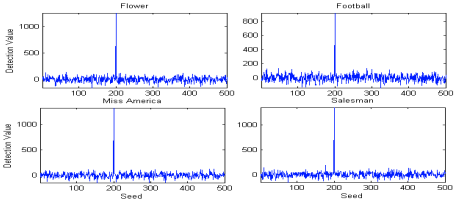


Fig. 8. Detection Value after additive Gaussian noise.

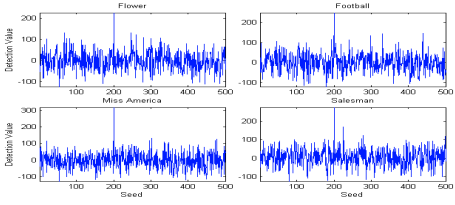


Fig. 9. Detection value after Low-pass filtering.

We added white Gaussian noise (variance = 1) to all frames of each sequence and detected the watermark. From the result shown in Figure 8, proposed algorithm shows robustness in reference to additive Gaussian noise. We applied low-pass filter (3×3) to all frames of each sequence and detected the watermark. Although low-pass filtering attenuates a specified range of high frequency components, robustness in reference to low-pass filtering was evident from the result shown in Figure 9. Median filter is a sort of low pass filter, so it removes the high frequency parts of the images. We applied median filter (3×3) to all frames of each sequence and detected the watermark.

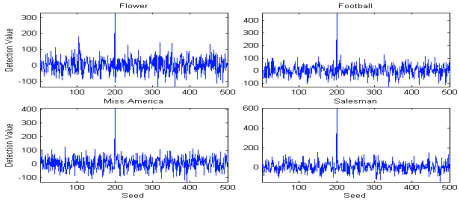


Fig. 10. Detection value after Median filter-Ing.

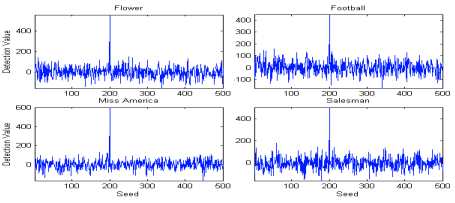


Fig. 11. Detection value after Histogram Equalization.

The results are similar to that of low-pass filtering attack. Figure 10 shows that the watermark is detected properly. The histogram of image represents the relative frequency of occurrence of the gray level image. Histogram Equalization is very useful in stretching the low contrast level of image by equalizing histogram of the original image. It was also evident that this algorithm is robust against histogram equalization from the result that was shown in Figure 11.

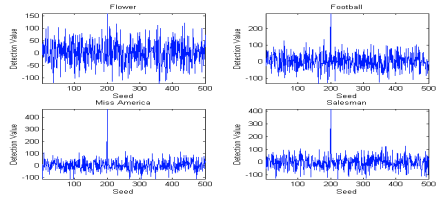


Fig. 12. Detection Value after MPEG re-encoding.

We also investigated the robustness of the watermark to MPEG re-encoding. After decoding the watermarked video data, we re-encoded the same watermarked video data with different bit-rate (0.8 Mbps). We see from Figure 12 that the watermark can be extracted and robust against this attack. From Figure 8, 9, 10, 11, and 12, we confirmed the robustness against the executed five attacks.

To confirm if proposed VWM algorithm can be applied for real time application, we computed the processing time that it takes to embed and detect the watermark and compared the processing time of the general MPEG video system with that of watermarking MPEG video system that embed and detect the watermark. Table 4 shows the ratio of processing time of general MPEG video system to that of watermarking MPEG video system.

Table 4. The ratio of a processing time.

		Flower	**Football**	**Miss A**	**Salesman**
Encoder		1.010	1.058	1.002	1.058
De-coder	Ran. Num.	1.134	1.142	1.109	1.131
	Text	1.016	1.016	1.015	1.016

From Table 4, we confirmed that there is no big difference between the processing times of two MPEG video systems. So to speak, the time required for watermark embedding is almost similar to that of the general encoding time, and the time required for watermark detection is about one eighth of general decoding time. The proposed algorithm has an advantage i.e. it is faster than existing real time VWM algorithms that were proposed by F. Hartung et al. [2] whose algorithm requires about one third of general decoding time to detect the watermark and G. C. Langelaar et al. [9] whose algorithm requires about one half of general decoding time to detect the watermark. So, proposed algorithm is suitable for real time VWM system. Moreover, if binary image and text is embedded instead of binary random numbers, processing time will be much more saved because it doesn't need to compute detection values of the watermarks with binary random numbers generated by any key numbers. We represent how many embedded bits are unchanged after four attacks. Table 5 shows the bit error rate that indicates how many bits have not been changed. Bit error rates were calculated

$$\frac{1}{2} - \frac{DV}{2 \times NEB} \tag{13}$$

where DV is detection value and NEB is the number of embedded bits.

Table 5. Bit error rate after attacks.

	Flower	Football	Miss.Ame.	Salesman
No attack	0.0	0.0	0.0	0.0
Add. Ga.	0.25	0.30	0.19	0.22
Lowpass	0.44	0.44	0.43	0.44
Median	0.42	0.38	0.40	0.36
Hist. Eq.	0.40	0.42	0.37	0.32
Re-encoding	0.46	0.42	0.35	0.40

From Table 5, the proposed VWM algorithm is more robust against the additive Gaussian noise attack than any other attack. In contrast with the additive Gaussian noise attack, other attacks have an impact on DCT coefficients but nevertheless proposed VWM algorithm is robust.

Table 6. Average number of Embedded bits in MPEG bitstream.

PictureType	Flower	Football	Miss A	Salesman
I (Diagonal)	1507.7	1559.4	1940.2	2101.7
I (VC or HC)	1030.9	400.7	116.8	540.9
I (Total)	2538.6	1960.1	2057	2642.6
P	109.1	288.5	29.5	109.6
B	6.0	62.4	1.4	7.3

While Table 3 shows the number of embedded bits in raw video data in the middle of MPEG encoding, Table 6 shows that of embedded bits in MPEG bitstream that has been encoded with 1.2Mbps. Comparing the two tables, it can be concluded that there is no big difference between two systems with respect to the number of embedded bits. Figure 13 shows how the number of bits increased in I picture, which is caused by watermark embedding, can be reduced in B and P pictures.

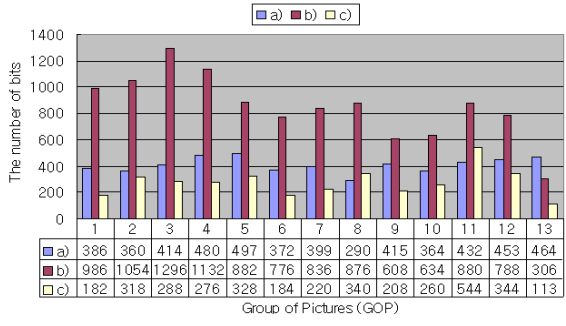


Fig. 13. a) Increased number of bits in I picture b) Possible reduction in number of bits with run-level (0,4) c) Possible reduction in number of bits with run-level (3,3) in case of "Flower Garden" MPEG bitstream.

4 Conclusions

This paper proposed the new watermarking scheme that is appropriate for real-time processing. By applying two different schemes according to Picture Type, the performance of the proposed VWM algorithm has been improved considerably. While previous real-time VWM algorithms have been generally weak against attacks, it has been proven that the proposed algorithm is robust against attacks and the processing time is reduced as well. Moreover computational complexity has been reduced because it does not need any preprocessing and extra computation except for computing VF and HF in embedding and detecting the watermark. The proposed VWM algorithm is also applicable to a real-time copy protection system for digital recoding devices. If the watermark exists in a compressed video, duplication is prohibited and its video can't be stored. If does not exist, duplication is allowed only one time and its video can be stored by an authenticated person. From the advantages of proposed algorithm shown in experimental results, the principle can be applied to both a real-time VWM system and a real-time copy protection system.

Acknowledgement. This work was supported by the Korea Research Foundation Grant (KRF-2003-041-D20470).

References

1. I. J. Cox, J. Kilian, F. T. Leighton and T. Shamoon, "Secure Spread Spectrum Watermarking for Multimedia," IEEE Trans. on Image Processing, vol. 6, no. 12, pp. 1673-1687, December. 1997.
2. F. Hartung and B. Girod, "Watermarking of Uncompressed and Compressed Video," Signal Processing 66, 283-301. 1998.
3. C. T. Hsu and J. L. Wu, "DCT-based Watermarking for Video," IEEE Trans. on Consumer Electronics, vol. 44, no. 1, February, 1998.
4. G. C. Langelaar, R. L. Lagendijk and J. Biemond, "Real-time Labeling Methods for MPEG Compressed Video," Proceeding of 18'th Symposium on Information Theory in the Benelux, Veldhoven, The Netherlands, May 1997.
5. G. Langelaar and R. L. Lagendijk, "Optimal differential Energy Watermarking of DCT Encoded Images and Video," IEEE Trans. on Image Processing, vol. 10, no. 1, pp. 148-158, Jan 2001.
6. F. W. Campbell and J. J. Kulikowski, "Orientation selectivity of the human visual system," in Journal Physiology, vol. 187, pp. 437-445, 1966.
7. M. D. Swanson, M. Kobayashi and A. H. Tewfik, "Multimedia Data-Embedding and Watermarking Technologies," Proceedings of the IEEE, vol. 86, no. 6, pp. 1064-1087, 1998.
8. D. Simitopoulos, S. A. Tsaftaris, N. V. Boulgouris, and M. G. Strintzis, "Compressed-domain video watermarking of MPEG streams," Proceedings Of 2002 IEEE International Conference on Multimedia and Expo, vol. 1, pp. 569 -572, 2002.
9. Chun-Shien Lu, Jan-Ru Chen, H.-Y. M. Liao, and Kuo-Chih Fan, "Real-time MPEG2 video watermarking in the VLC domain," Proceedings Of 16th International Conference on Pattern Recoginition, vol. 2, pp. 552-555, 2002.

A TCP-Friendly Congestion Control Scheme Using Hybrid Approach for Reducing Transmission Delay of Real-Time Video Stream

Jong-Un Yang[1], Jeong-Hyun Cho[1], Sang-Hyun Bae[2*], and In-Ho Ra[1]

[1] School of Electronic & Information Engineering,
Kunsan National University, Kunsan, Korea
{bedroses, ihra, cd20}@kunsan.ac.kr

[2] Dept. of Computer Science & Statistic, College of Natural Science,
Chosun University, Gwangju, Korea
shbae@chosun.ac.kr

Abstract. Recently, due to the high development of the Internet, the need for multimedia streams such as digital audio and video is increasing more and more. In the case of transmitting multimedia streams using the User Datagram Protocol (UDP), it may cause starvation of TCP traffic on the same transmission path, thus resulting in congestion collapse and an enormous delay because UDP does not perform TCP-like congestion control. Because of this problem, diverse research is being conducted on new transmission schemes and protocols intended to efficiently reduce the transmission delay of real-time multimedia streams and perform congestion control. The TCP-friendly congestion control schemes can be classified into the window-based congestion control, which uses the general congestion window management function, and the rate-based congestion control, which dynamically adjusts transmission rate by using TCP modeling equations. In this paper, we suggest the square-root congestion avoidance algorithm with the hybrid TCP-friendly congestion control scheme in which the window-based and rate-based congestion controls are dealt with in a combined way. We apply the proposed algorithm to the existing TEAR. We simulate the performance of the proposed TEAR by using NS, and the result shows that it gives better improvement in respect to stability needed for providing congestion control than the existing TEAR.

- This work was done as a part of Information & Communication fundamental Technology Research Program supported by Ministry of Information & Communication in republic of Korea.
- * Corresponding Author: shbae@chosun.ac.kr

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 359–368, 2004.
© Springer-Verlag Berlin Heidelberg 2004

1 Introduction

According to the recent development of information and telecommunication technology and the Internet, demands for various multimedia services have increased greatly so that high transmission bandwidth and low transmission delay are being required more and more for real-time transmission of multimedia streams more and more. For supporting real-time multimedia streaming to handle the problem, a transmission mechanism must be designed in great consideration of such conditions as packet loss, transmission delay and bandwidth.

In this paper, we study a scheme for controlling any traffic congestion, which would be experienced from the characteristics of existing transport protocols such as TCP or UDP during real-time transmission of a large amount of streaming data such as video, and audio streams. Because UDP, which is a connectionless transmission protocol, does not perform error control or flow control, it is often used in multimedia applications that require faster switching on network nodes. When UDP is used to transmit multimedia streams, unlike TCP, it does not perform congestion control and thus monopolizes bandwidths, which leads to starvation of TCP traffic on the same transmission path, resulting in congestion collapse. It often happens when the channel bandwidth allocated to UDP increases compared with that of TCP, which performs congestion control [1]. Similarly, when TCP congestion control is applied to real-time multimedia applications, the service quality of multimedia data is greatly decreased. This problem is caused by rapid changes in the transmission delay when the data transmission rate at a sender site is increased or decreased highly. In order to resolve this problem, various studies have been conducted on TCP-friendly congestion control schemes suitable for real-time multimedia data transmission.

In general, TCP-friendly congestion control schemes can be classified into two parts: one is the window-based congestion control, which performs congestion control by using the congestion window size management function, and the other is the rate-based congestion control, which dynamically adjusts the transmission rate by using TCP modeling equations.

In this paper, we propose a hybrid-type TCP-friendly congestion control scheme where the delay-based square-root congestion avoidance scheme is applied for providing more stable transmission of multimedia data through a shared network environment such as the Internet. The rest of this paper is as follows: In Section 2, we describe the features and methods of the TCP-friendly congestion control schemes. In Section 3 and 4, we present a hybrid TCP-friendly congestion control scheme to which the suggested scheme is applied and show the results of the performance evaluation with NS-2. Finally, in Section 5 we conclude the paper with a summary of the points raised throughout this work.

2 TCP-Friendly Congestion Control Schemes

2.1 Window-Based Congestion Control

The window-based congestion control scheme is designed to adjust the transmission rate by managing the congestion window. A sender can transmit packets with up to the maximum window size and then the receiver sends ACK for each received packet back to the sender, thus the ACK adjusts the sender's transmission rate. That is to say, the sender transmits packets at ACK intervals. This function of the window-based congestion window is called 'ACK-clocking'. Types of the window-based congestion control schemes known up to now are AIMD (Additive Increase Multiplicative Decrease), GAIMD (General Additive Increase Multiplicative Decrease) and so on.

AIMD Congestion Control. The sender increases the sizes of a congestion window gradually to use available bandwidth while it decreases the size by half when a packet loss, which is regarded as a congestion indicator, occurs. The AIMD is given by

$$\begin{aligned} I &: w_{t+R} \leftarrow w_t + \alpha & \alpha &> 0 \\ D &: w_{t+\delta t} \leftarrow (1-\beta)\omega & 0 &< \beta < 1 \end{aligned} \tag{1}$$

Where I refers to an increase in the size of the congestion window while D represents a decrease in its size during the RTT. In general, it is set to $\alpha = 1$, $\beta = 1/2$ in TCP. With the AIMD scheme, it is possible to use available bandwidth effectively and be applied to most applications [2].

GAIMD Congestion Control. GAIMD is a generalized one after extending of the AIMD scheme. In order for GAIMD to be TCP-friendly, α and β must be given by

$$\alpha = \frac{4(1-\beta^2)}{3} \tag{2}$$

Generally, it is set to $\beta = 7/8$ for smoothing transmission of multimedia applications. Thus, α is set to the value 0.31 by the equation (2) [3].

2.2 Rate-Based Congestion Control

The rate-based congestion control scheme dynamically adjusts the transmission rate by using a network feedback function for informing that congestion is occuring to the sender. TFRC (TCP-Friendly Rate Control), which uses a TCP modeling equation, is one of the rate-based congestion control schemes.

TCP Modeling equation. The TCP throughput is defined by

$$T = \frac{s}{t_{rtt}\sqrt{\frac{2p}{3}} + t_{rto}\min(1.3\sqrt{\frac{3p}{8}})p(1+32p^2)} \tag{3}$$

Where t_{rtt} is the round-trip time (RTT), t_{rto} is the retransmit timeout, s is packet size, and p is the packet loss event rate.

TFRC. TFRC is a scheme which adjusts the transmission rate using a TCP modeling equation. To perform rate-based congestion control with the equation (3), it is important to measure the packet loss rate correctly. TFRC measures the packet loss event rate with the average loss interval and EWMA. The TFRC receiver updates and sends the parameters for the TCP equation to the sender per each RTT and the sender transmits packets at the transmission rate obtained by calculation using the updated information.

TFRC uses delay-based congestion avoidance to improve the performance of the protocol. It can maintain a relatively stable transmission rate by sufficiently responding to traffics competing with one another for bandwidths [5], [6].

3 Hybrid TCP-Friendly Congestion Control Scheme

TEAR (TCP emulation at receivers) is a representative protocol belonging to the hybrid TCP-friendly congestion control schemes [6], [7]. Based on this, we propose a square-root congestion avoidance algorithm for improving the performance of the existing TEAR.

3.1 TEAR Introduction

The TEAR receiver manages the size of the congestion window and calculates the throughput with it. Next, it transmits the calculated throughput to the sender through a feedback function. After this, the sender determines the transmission rate based on the feedback information.

Increase window algorithm. When a packet is received in the state of SLOW-START, CONGESTION-AVOIDANCE, the congestion window size is increased by the operation as shown in Fig 1.

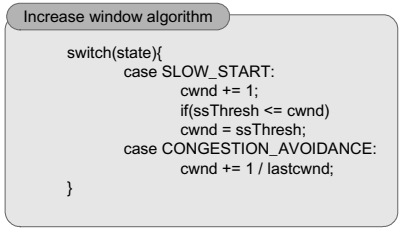


Fig. 1. Increase window algorithm

Decrease window algorithm. When a packet loss occurs, it results in a state transition from the SLOW-START, CONGESTION-AVOIDANCE state to the GAP state. The size of the congestion window does not change in the GAP state, which is a relay state where the receiver determines whether the packet loss is caused by timeout or by three duplicate ACKs. Fig.2 shows the algorithm for decreasing the congestion window.

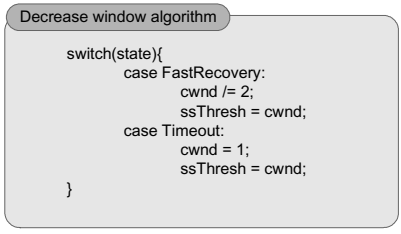


Fig. 2. Decrease window algorithm

3.2 A Delay-Based Square-Root Congestion Avoidance Algorithm for Improving the Performance of TEAR

The TEAR receiver calculates the transmission rate in a unit of epochs instead of every RTT. It is necessary to prevent any transmission rate from high fluctuation in a saw-tooth shape. An epoch is defined as the time between the two successive rate decreasing events. Fig.3. shows epochs. In order to prevent any unnecessary rate fluctuations caused by noise, a weighted average for the rate of the last W epochs is used to calculate the smoothing throughput and the result should be sent to the sender through feedback. With this information, the sender can adjust the transmission interval between packets.

In this paper, we propose a scheme for evaluating the transmission interval between packets, $t_{\text{inter-packet}}$, as shown in the equation (4) by adopting the delay-based square-root congestion avoidance scheme, where the proposed scheme can be applied well to diverse network environments including the Internet.

$$t_{\text{inter-packet}} = \frac{S \times RTT_{\text{current}}}{T_{\text{estimated}} \sqrt{SRTT}} \tag{4}$$

Here, $T_{\text{estimated}}$ is the throughput calculated by the receiver, and S is the packet size. The proposed scheme enables itself to adapt to a shared network environment such as the Internet by performing congestion avoidance on the basis of network delay, which is the best method to predict the current network state. In other words, if the current network delay is increasing more than the previous one, the current network state is likely to fall into congestion. In such a case, the transmission interval is lengthened in order to avoid congestion, but it is shortened to maintain stable TEAR performance. Fig .4 shows the overall operating mechanism of the proposed scheme.

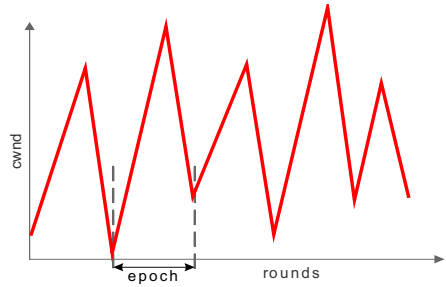


Fig. 3. Epoch

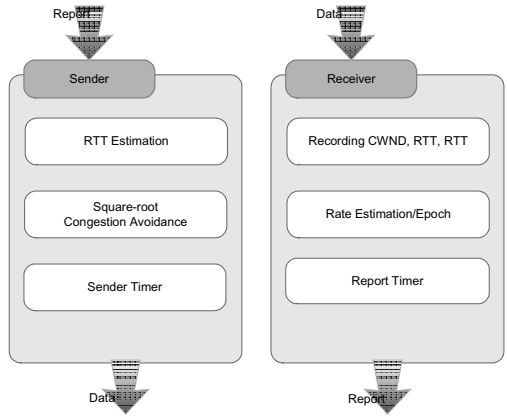


Fig. 4. Overall operating mechanism

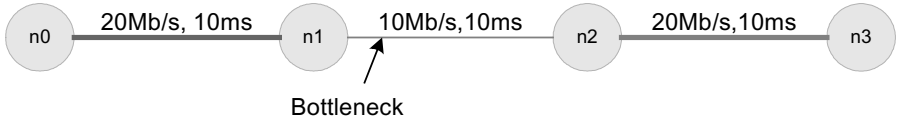


Fig. 5. Simulation topology

4 Simulation Results

We simulate the performance of the proposed scheme in a topological environment shown in Fig. 5 with NS-2[8]. For comparisons of the performance between the proposed TEAR, TCP and the existing TEAR, we measure the aggregate and instantaneous rate of them under different conditions. We also perform comparative analysis on the performance between the existing TEAR and the proposed TEAR under the same conditions (Feedback : 1 RTT, DropTail, TCP-SACK).

From Fig. 6 to 9, they show the simulation results with the same conditions on the existing TEAR known up to now. Fig. 6 shows a comparison of the aggregate rate of a TCP flow with that of a proposed TEAR flow. Similarly, Fig. 8 shows a comparison of the aggregate rate of TCP and that of TEAR, where the number of TCP flows was increased to two. Fig. 7 shows a comparison of the instantaneous rate of TCP with that of TEAR. Likewise, Fig. 9 shows a comparison of the instantaneous rate of two TCP flows with that of the proposed TEAR flow.

Finally, Fig. 10 and 11 respectively show the result of the comparison of the instantaneous rate of TCP with that of the proposed TEAR. With these results, we find that the proposed scheme can be used in various types of shared networks by applying a method that is delay-based in various types of shared networks by applying a method of delay-based square-root congestion to the existing TEAR. And we also expect that the proposed TEAR can be properly used in multimedia applications, where it is required for real-time transfer of continuous media without jitter or skew.

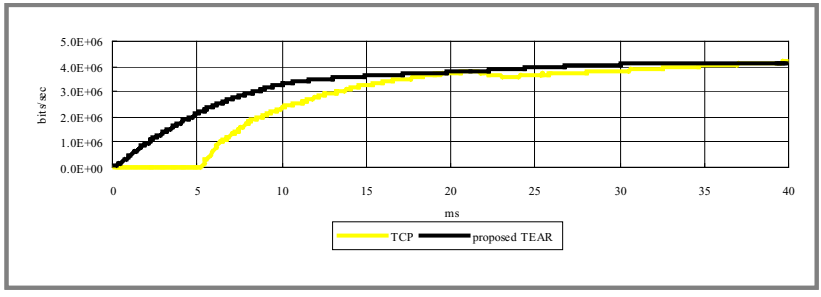


Fig. 6. Comparison of TCP(1 flow) with the proposed TEAR(1 flow) on an aggregate rate

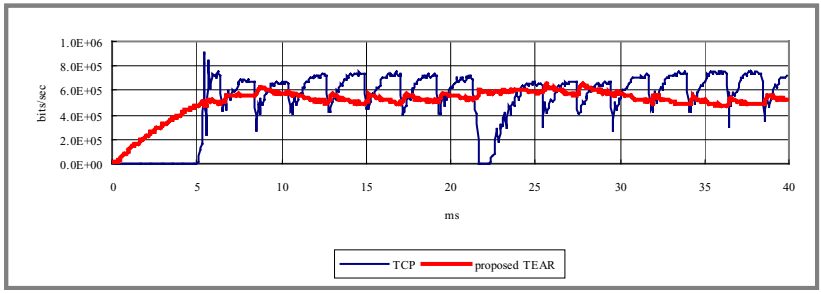


Fig. 7. Comparison of TCP(1 flow) with the proposed TEAR(1 flow) on an instantaneous rate

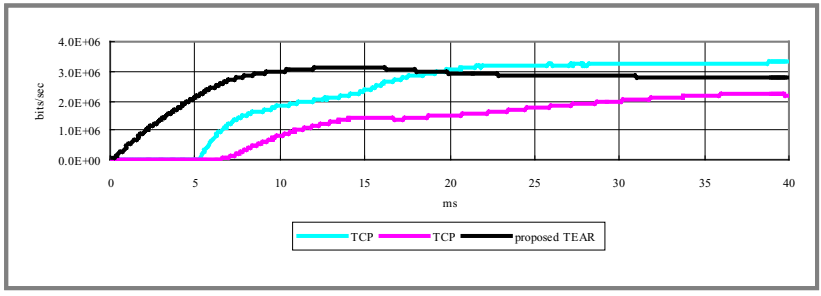


Fig. 8. Comparison of TCP (2 flows) with the proposed TEAR (1 flow) on an aggregate rate

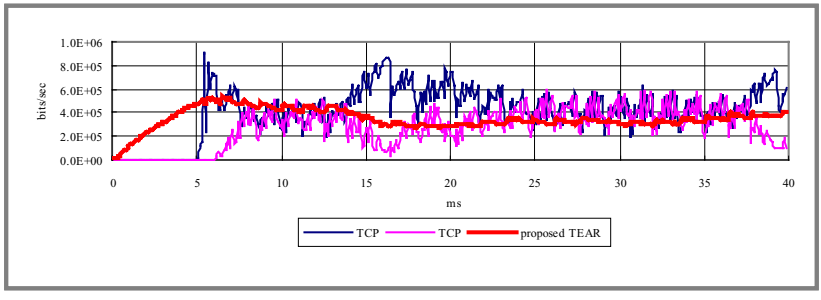


Fig. 9. Comparison of TCP(2 flows) with the proposed TEAR (1 flow) on an instantaneous rate

5 Conclusion

In this paper, we propose a hybrid TCP-friendly congestion control scheme using a delay-based square-root congestion control method for providing synchronous real-

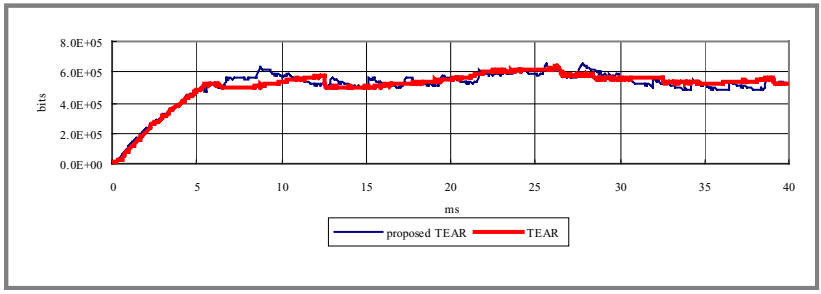


Fig. 10. Comparison of the existing TEAR (1 flow) with the proposed TEAR(1 flow) on an instantaneous rate when competing TCP (1 flow)

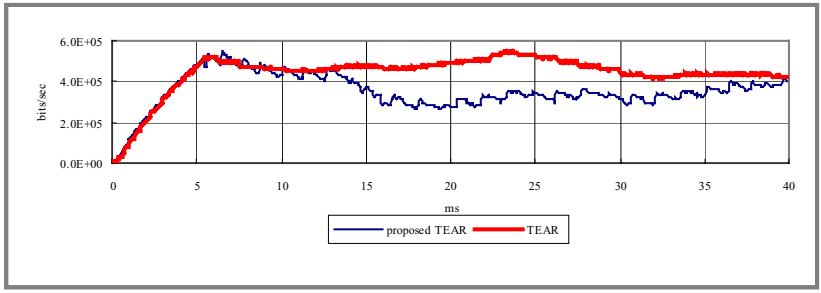


Fig. 11. Comparison of the existing TEAR(1 flow) with the proposed TEAR(1 flow) on an instantaneous rate when competing TCP (2 flows)

time transmission of multimedia using the congestion avoidance technique. The proposed scheme is designed to make it feasible to use the network bandwidth fairly in the transport layer even in the case of competing with TCP flows. And it also gives a method to avoid a rapid degradation of quality of services of multimedia application, which is caused by high fluctuations in transmission rate. With this, a smoothing rate can be supported by calculating the transmission rate based on the epoch units. Finally, we improve the performance of existing TEAR. With the simulation results, we show that the proposed scheme can be stably used in shared network environments such as the Internet. And it also shows that our scheme improves the ability of congestion avoidance better than the existing TEAR.

References

1. S. Floyd, K. Fall., “Promoting the use of End-to-end Congestion Control in the Internet”, IEEE/ACM Transactions on Networking, Aug 1999.
2. S. Jin, L. Guo, I. Matta, A. Bestavros, “A Spectrum of TCP-friendly Window-based Congestion Control Algorithms”, July 2002.

3. Y. Richard Yang, Simon S. Lam, “General AIMD Congestion Control”, In Proceedings of ICNP, November 2000.
4. D. Bansal, H. Balakrishnam, “Binomial Congestion Control Algorithms”, In Proceedings of IEEE INFOCOM, April 2001.
5. S. Floyd, M. Handley, J. Padhye, J. Widmer, “Equation-based Congestion Control for Unicast Applications”, Technical Report ACIRI, Feb 2000.
6. J. Widmer, R. Denda, M. Mauve, “A Survey on TCP-Friendly Congestion Control (Extended version)”, Feb 2001.
7. I. Rhee, V. Ozdemir, Y. YI., “TEAR : TCP Emulation At Receivers - Flow Control for Multimedia Streaming”, Technical Report NCSU, April 2000.
8. Network Simulator - ns-2, http:// www.isi.edu/nsnam/ns

Object Boundary Edge Selection Using Level-of-Detail Canny Edges

Jihun Park[1] and Sunghun Park[2]

[1] Department of Computer Engineering
Hongik University,
Seoul, Korea
jhpark@hongik.ac.kr
[2] Department of Management Information Systems
Myongji University,
Seoul, Korea
shpark@mju.ac.kr

Abstract. Recently, Nguyen proposed a method[1] for tracking a non-parameterized object (subject) contour in a single video stream with a moving camera and changing background. Nguyen's approach combined outputs of two steps: creating a predicted contour and removing background edges. Nguyen's background edge removal method of leaving many irrelevant edges is subject to inaccurate contour tracking in a *complex* scene. Nguyen's method[1] of combining the predicted contour computed from the previous frame accumulates tracking error. We propose a brand-new method for tracking a nonparameterized subject contour in a single video stream with a moving camera and changing background. Our method is based on level-of-detail (LOD) Canny edge maps and graph-based routing operations on the LOD maps. We compute a predicted contour as Nguyen do. But to reduce side-effects because of irrelevant edges, we start our basic tracking using *simple (strong)* Canny edges generated from large image intensity gradients of an input image, called *Scanny* edges. Starting from *Scanny* edges, we get more edge pixels ranging from simple Canny edge maps untill the most detailed (weaker) Canny edge maps, called *Wcanny* maps. If *Scanny* edges are disconnected, routing between disconnected parts are planned using level-of-detail Canny edges, favoring stronger Canny edge pixels. Our accurate tracking is based on reducing effects from irrelevant edges by selecting the strongest edge pixels only, thereby relying on the current frame edge pixel as much as possible contrary to Nguyen's approach of always combining the previous contour. Our experimental results show that this tracking approach is robust enough to handle a complex-textured scene.

1 Introduction and Related Works

Tracking moving subjects is a hot issue because of a wide variety of applications in computer vision such as video coding, video surveillance, monitoring, augmented reality, and robotics. This paper addresses the problem of selecting boundary edges for robust contour tracking in a single video stream with a

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 369–378, 2004.
© Springer-Verlag Berlin Heidelberg 2004

moving camera and changing background. We track a highly textured subject moving in a complex scene compared to a relatively simple subject tracking done by others. We mean complex because both tracked subject and background scene leave many edges after edge detection.

We can classify the methods of representing an object contour into two categories depending on the method used; parameterized contour or nonparameterized contour. In tracking a parameterized contour, an object contour estimating the motion is represented by using parameters. These methods use Snake model[2] in general; Kalman Snake[3] and Adaptive Motion Snake[4] are popular Snake models. In the method of tracking a nonparameterized contour, an object contour as an object border is represented. The contour created by these algorithms is represented as a set of pixels. Paragios's algorithm[5] and Nguyen's algorithm[1] are popular in these approaches.

In Nguyen's algorithm[1], a watershed line that was determined by using the watershed segmentation[6] and the watershed line smoothing energy[7,1] becomes the new contour of a tracked object. Nguyen's approach removed background edges by using object motion. But Nguyen's approach left many irrelevant edges that prohibit accurate contour tracking because removing background edges are difficult. The watershed line is generated by combining the previous frame contour and the current frame Canny edges that do not always make a closed edge contour. In this way, tracking errors are accumulated by always including the previous contour regardless of the intensity of the current Canny edges. Predicted contour computed from the previous frame is usually different from the exact contour for the current frame. Usually a big change between the previous and current contour shapes makes this kind of contour tracking difficult.

To overcome Nguyen's two problems, difficulty in removing noisy background edges and accumulating tracking errors, we proposes a new method to increase subject tracking accuracy by using level-of-detail (LOD) Canny edge maps. We compute a predicted contour as Nguyen does. But to reduce side-effects because of irrelevant edges, we start our basic tracking contour using *simple (strong)* Canny edges generated from large image intensity gradients, called *Scanny* edges. Our new method selects only the Canny edges with large image intensity gradient values, *Scanny* edges. A *Scanny* edge map does not have noisy background edges, and looks simple, meaning less edges in the Canny edge map of the scene. Our accurate tracking is based on reducing effects from irrelevant edges by selecting strongest edge pixels only, relying on the current frame edge pixel as much as possible contrary to Nguyen's approach of always combining the previous contour. For Canny edge maps generated with smaller image intensity gradient values, we call $Wcanny_i$, $i = 1, \cdots, N$ where N is the number of level-of-detail of Canny edge maps. $Wcanny_1$ has the simplest Canny edges generated from a set of large *(strongest)* intensity gradient value edges. $Wcanny_N$ has the most detailed Canny edges generated by an accumulation from largest (*strongest*) till smallest (*weakest*) intensity gradient valued edges. Basically, we rely only on *Scanny* to find a basic (but not closed) tracked contour frame, and then we seek additional edge pixels from $Wcanny_i$s according to the descending sequence of multi-level detailed edge pixels, following LOD in edge maps. We consider *Scanny*

edges around a predicted contour, computed from the previous frame contour, to be likely a part of the new contour. To make a closed contour including the above partial contour frame made from *Scanny* edges, we check $Wcanny_i$ edge maps in the order of LOD. We do a routing, between two disconnected *Scanny* edge pixel, using LOD *Wcanny* edge maps favoring stronger edge maps. The disconnected contour is connected using Dijkstra's minimum cost routing. Our accurate tracking is based on reducing effects from irrelevant edges by selecting strongest Canny edges using the LOD Canny edge maps starting from the *simple* Canny edge map. Our experimental results show that our tracking approach is robust enough to handle a highly textured scene.

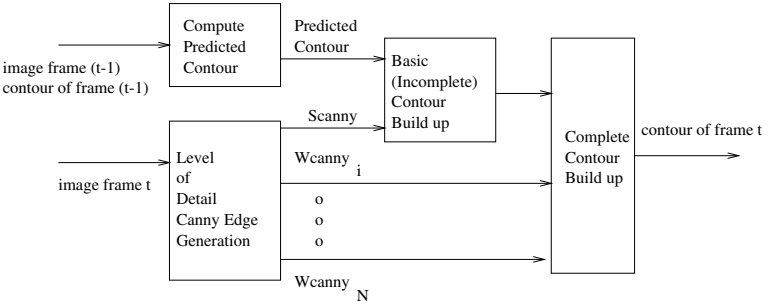
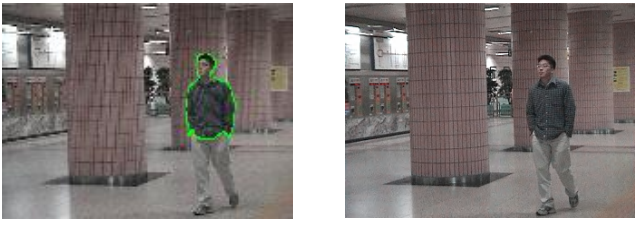


Fig. 1. Overview of Our Single Frame Tracking

2 Overview of Our System

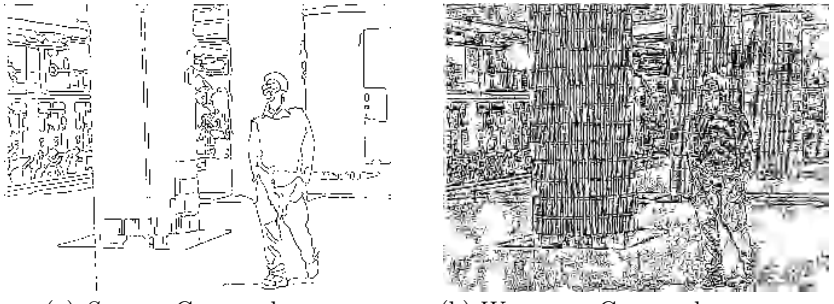
Figure 1 shows an overview of our system to track a single image frame. As inputs, we get a previous image frame, denoted as frame $(t-1)$, and its corresponding tracked subject contour $(t-1)$, and a current image frame, denoted as frame t. From frame $(t-1)$, contour $(t-1)$, and frame t we can compute a predicted contour for frame t. From input frame t, we generate various detailed levels of Canny edge image maps for the input frame t. We select *Scanny* edges from the LOD Canny edge maps. From a *Scanny* edge map, we derive a corresponding distance map. Using the predicted contour, we find the best matching between the predicted contour and the distance map. *Scanny* edge pixels matching with the predicted contour become the frame of the contour build up. We call these pixels as *selected Scanny contour pixels*. *Selected Scanny contour pixels*, generated using *Scanny* and predicted contour, is the most reliable (but not closed) contour pixels to start building a closed tracked contour, and they are stored in the *selected Scanny found list*. To build a closed contour for the frame (t), we use LOD Canny edge maps around the predicted contour. The LOD Canny edge maps consist of various levels in Canny edge generation. The *Scanny* edge pixels are assigned LOD value one, LOD value two for $Wcanny_1$

edge pixels, and LOD value $(N+1)$ for $Wcanny_N$. LOD value 255 is reserved for pixels with no edge. We route a path to connect adjacent *selected Scanny contour pixels* in the found list. If we finish connecting every adjacent *selected Scanny contour pixel* pairs, we get a basic contour although it is not guaranteed to be the best. We mean *best* because the contour is four-neighbor connected and follows every possible *Scanny* edge. We run a final routing using the computed basic contour and *Scanny* edges around it to find the best contour. The resulting contour becomes the contour of frame (t). Figure 2(a) is an input frame $(t-1)$ and its corresponding tracked subject contour $(t-1)$, while Figure 2(b) is an input frame t.



(a) frame $(t-1)$ with contour (b) frame t

Fig. 2. Input images and contour for tracking



(a) *Scanny* Canny edge map (b) $Wcanny_N$ Canny edge map

Fig. 3. *Scanny* and $Wcanny_N$ Canny edge maps

3 Generating Level-of-Detail Canny Edge Maps and Edge Pixel Selection

3.1 Generating Level-of-Detail Canny Edge Maps

By varying control parameters, we can get various Canny edge maps given a single image. The resulting Canny edge maps are mainly affected by the image

intensity changes between pixels. We take advantage of the fact that we can get various Canny edge maps by varying these control parameters. Usually very detailed Canny edge maps confuse us to find the exact outline, but simple Canny edge maps generated from large image intensity changes do not have enough detail to make a closed contour for the tracked subject. But simple Canny edge maps are very reliable because they are generated only if there are big intensity changes in the image. We need both simple and detailed Canny edge maps (ranging from simplest untill most detailed Canny edge maps) for the best subject tracking. We generate various detailed Canny edge maps varying values of control parameters. We order the resulting Canny edge maps according to the detailed level. Then, we take top 10 percent of simple Canny edge maps and union them into pixel-level to make a *Scanny* edge map. The rest of Canny edge maps are used to generate $Wcanny_i$. $Wcanny_1$ is a pixel-wise union of *Scanny* and next detailed sets of Canny edge maps. $Wcanny_i$ is generated by unioning $Wcanny_{(i-1)}$ and next detailed sets of Canny edge maps and so on. $Wcanny_N$ has the union of all levels of detail Canny edges generated by an accumulation from *highest-to-lowest* intensity gradient valued edges. Figure 3 shows an example of *Scanny* and $Wcanny_N$ Canny edge maps.

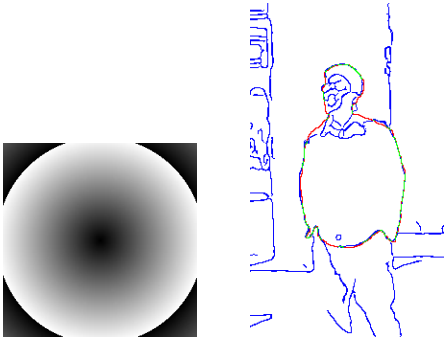
(a) (b) (c)

Fig. 4. Predicted contour from frame $(t-1)$(a), distance map generated from *Scanny*(b), matching between predicted contour and *Scanny* distance map(c)

3.2 Strong Canny Pixel Selection

Nguyen[1] removed background edges using object motion. But, Nguyen's approach left many irrelevant edges in the following cases: 1) an edge segment that has the same direction as object motion and length that exceeds the length of object motion, and 2) inner edges of a tracked object. These irrelevant edges prohibit accurate contour tracking.

We do not remove any background edges, and removing background edges are not easy. One of the reason to remove background edges is because they are very noisy and disturbs correct subject tracking. Rather than removing background



(a) (b)

Fig. 5. Circular distance map used in matching(a), matching with local weight(b)

edges, we start with *Scanny* edge map as presented in Figure 3(a), that has simple edges in a scene. By using an image matching as Nguyen did[1], we can get a predicted contour as presented in Figure 4(a). Then, we generate a distance map of *Scanny* as in Figure 4(b). Given a pixel on the predicted contour, we find the corresponding *Scanny* edge pixel, if one exists, by matching between the predicted contour and the distance map. If the matching point corresponds to a *Scanny* edge pixel, then the pixel is selected. We call this pixel as a *selected Scanny contour pixel*. In this matching, we may use a circular distance map as presented in Figure 5(a) to give more weight for the *local* matching. The center of the circular distance map is positioned at the reference pixel on the predicted contour. Figure 5(b) shows a collection of best matching with the reference contour pixel point (marked as red cross). The green contour denotes the predicted contour, while black edge pixels denote *Scanny* edge pixels. Gray levels are because of a distance map of *Scanny* edge map. Green pixel denotes matching on *Scanny* pixel, and red pixel denotes a close matching. Green pixels are stored in the found list of *selected Scanny contour pixel*. From the matching along the predicted contour, we get a *found list* of *selected Scanny contour pixels*. These pixels are usually not connected as a four neighbor connection, but these pixels most likely become part of the new contour to be computed.

3.3 Strong Canny Contour Pixel Connection for a Basic Contour Building

We need to connect adjacent *selected Scanny contour pixels*, stored in the *found list*, to build a new closed contour. We mean adjacent, adjacent in the *found list*. This computed contour will be the *basic* tracked subject contour for $frame\ (t)$. Our $Wcanny$ Canny edge tracing to find a route to connect *selected Scanny contour pixels* is done using the concept of LOD. The LOD Canny edge maps consist of various levels in Canny edge generation. The *Scanny* edge pixels are assigned LOD value one, LOD value two for $Wcanny_1$ edge pixels, and LOD value $(N + 1)$ for $Wcanny_N$ and so on. LOD value 255 is reserved for pixels

with no edge. We take a part of the LOD Canny edge map around two adjacent *selected Scanny contour pixels*. Pixels of the LOD map become nodes, and we determine weights between *adjacent* pixels. We mean *adjacent* pixels to be four-neighbor connected pixels. We determine weights between adjacent pixels using a Canny edge LOD value of each pixel. We favor traversing the most simple (stronger) edge pixels in the map rather than the most detailed (weaker) edge pixel in level-of-detail. We assign the lowest weight between two adjacent *Scanny* edge pixels to encourage *Scanny*-based routing. The weight function guarantees to take stronger Canny edges in the optimum path routing. If there is no edge pixel present, the routing takes ordinary pixels with LOD value 255 to make a closed contour. The routing is done using Dijkstra's minimum cost routing algorithm. We route a path to connect each adjacent *selected Scanny contour pixels* pair in the *found list*. If we finish connecting all adjacent *selected Scanny contour pixels* pairs, we get a basic contour although it is not guaranteed to be the globally best, but this guaranteed to be locally best. We mean *global* considering the entire contour rather than considering a part of the edge map. To get a *globally best* contour, we mean *best* that the contour is four-neighbor connected and follows every possible *Scanny* edges, we run a final routing using the computed basic contour and *Scanny* edges around the computed contour. The resulting contour becomes the contour of $frame\ (t)$. For the final contour routing, our routing node (pixel) consists of *Scanny* pixels as well as the computed contour pixels, the pixels found from the routing between *selected Scanny contour pixels*, and each node has two kinds of classes each for *Scanny* and the computed contour pixels. We assign lowest cost between adjacent *Scanny* pixels, while higher cost between pixels of the computed basic contour and highest cost between no adjacent pixels. This has an effect of favoring *Scanny* edges rather than computed contour pixels. If there is no route made by *Scanny* pixels for a part of an edge map, then a corresponding part of the computed contour is selected.

4 Experimental Results

We have experimented with easily available video sequences generated with a home camcorder, SONY DCR-PC3. We have generated 304 different LOD Canny edge maps, and union simplest 30 (top 10 percent) Canny edge maps to make *Scanny* Canny edge map. We have 274 $Wcanny_i$s. Although 274 $Wcanny$s does not take long computational time to be generated, it is not necessary to keep 274 different levels. Although 304 Canny edge maps do not take a long computational time to be generated, it is not necessary to keep 304 different levels. We may vary the percentage of Canny edge maps in determining a *Scanny* edge map, but the percentage is not critical in tracking performance as far as we take 10 to 30 percent of the Canny edge maps.

Figure 6 shows a man walking in a subway hall. The hall tiles as well as a cross stripe shirt generate many complicated Canny edges. The tracked contour shape and color changes as the man with a cross stripe shirt rotates from facing

Fig. 6. Tracking result at frame 0, the input frame with input contour(a), at frame 15(b), at frame 30(c), at frame 45(d), at frame 60(e), at frame 75(f), at frame 90(g), at frame 105(h), at frame 120(e), at frame 135(j), at frame 140, before occluding a woman(k), at frame 147, after occlusion(l), at frame 150(m), at frame 165(n), at frame 180(o), at frame 185, before occluding a second woman(p), at frame 194, after occluding the second woman(q), at frame 195(r), at frame 210(s), at frame 211, before occluding a man(t), at frame 214, after occluding the man(u), at frame 225(v), at frame 240(w), at frame 255(x)

the front to the back as he comes closer to a camera and then moves away from it. There are many edge pixels in the background and the subject has many edges inside the tracked contour. There are other people moving in different directions, in the background. To make tracking more difficult, the face color of the tacked subject is similar to the hall wall color (Figure 6(b,d,e,h,i)) while his shirt color is similar to that of stairs (Figure 6(n–x)), and tracked body black hair is interfered with by a walking woman in Figure 6(p–r), and by a man with a black suit in Figure 6(s–u). Our tracked contour is bothered by these interferences, but recovers as soon as we get *Scanny* edges for the interfered part unless background *Scanny* edges interfere. Even under this complex circumstance, our boundary edge-based tracking was satisfactory.

5 Conclusion

In this paper, we proposed a brand-new method of improving accuracy in tracking a highly textured subject. We start by selecting a boundary edge pixel from the simple (strong) Canny edge map, referring to the most detailed edge map to get edge information along the level-of-detail Canny edge maps. Our basic tracking frame is determined from the strong Canny edge map and the missing edges are filled by the detailed Canny edges along the LOD hierarchy. If no edge is present, then ordinary pixel will be selected based on Dijkstra's routing algorithm. Even though detailed Canny edges are noisy, our basic tracking frame is determined from the *Scanny*, the strong Canny edge map, and is not disturbed by noisy edges. This has an effect of Nguyen's background noisy edge removal. If there is no edge information available because of the same color with background, our tracking performance degrades heavily, and this is inevitable for all approaches. But our tracking performance recovers whenever we get edge information back and there is no background strong Canny edges confusing the tracked contour.

By using our novel method, our computation is not bothered by noisy edges or small cross stripe textures, resulting in a robust tracking. Our experimental results show that our tracking approach is reliable enough to handle a sudden change of the tracked subject shape in a complex scene.

Acknowledgements. The video sequence used in this paper was provided by Taeyong Kim, Korea University, Seoul, Korea. This research was supported by the 2004 Hongik University Academic Research Support Fund.

References

1. Nguyen, H.T., Worring, M., van den Boomgaard, R., Smeulders, A.W.M.: Tracking nonparameterized object contours in video. IEEE Trans. Image Processing **11** (2002) 1081–1091
2. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. International Journal of Computer Vision **1** (1987) 321–331

3. Peterfreund, N.: Robust tracking of position and velocity with kalman snakes. IEEE Trans. P.A.M.I. **21** (1999) 564–569
4. Fu, Y., Erdem, A.T., Tekalp, A.M.: Tracking visible boundary of objects using occlusion adaptive motion snake. IEEE Trans. Image Processing **9** (2000) 2051–2060
5. Paragios, N., Deriche, R.: Geodesic active contours and level sets for the detection and tracking of moving objects. IEEE Trans. P.A.M.I. **22** (2000) 266–280
6. Roerdink, J.B.T.M., Meijster, A.: The watershed transform: Definition, algorithms and parallelization strategies. Fundamenta Informaticae **41** (2000) 187–228
7. Nguyen, H.T., Worring, M., van den Boomgaard, R.: Watersnakes: energy-driven watershed segmentation. IEEE Trans. P.A.M.I. **25** (2003) 330–342

Inverse Dithering through IMAP Estimation

Monia Discepoli[1,2] and Ivan Gerace[1]

[1] Dipartimento di Matematica e Informatica, Università degli Studi di Perugia, via Vanvitelli 1, I-06123 PG, Italia.
[2] Dipartimento di Matematica "Ulisse Dini", Università degli Studi di Firenze, viale Morgagni 67a, I-50134 FI, Italia.

Abstract. In this paper we extend the use of the IMAP (Indirect Maximum A Priori) estimation to the inverse dithering problem. To find a continuous grey-levels image from a dithered image we have to estimate an appropriate blur mask. By the IMAP estimation we find the blur mask that permits to obtain the image that better fits the a priori knowledge on it. To reduce the computational cost we use an E-GNC (Extended Graduated Non-Convex) algorithm to perform the image restoration during the estimation.

Keywords: Inverse Dithering, Regularization, IMAP Estimation, GNC Algorithms.

1 Introduction

Spatial dithering is often referred as *digital halftoning*. This is a method to obtain a black-white image given the illusion to observe an image with 256 grey-levels. In practice the main advantage of this technique is the reduction of the number of data to transmit or to store the image. Digital halftoning is the basic idea of many techniques used to reduce the number of colors of the involved image [1, 22,24,25]. Digital halftoning has a lot of practical applications, for example in the use of displays and laser printers, indeed in this case it is necessary for the visualization of grey-levels image to binary output devices.

When a human being looks at an image he/she has the illusion of continuous-tone, due to the fact that human eyes average the grey-levels in a neighborhood of the observed points [21]. The problem of inverse dithering consists in estimating both the original grey-levels image and the blur mask that best fits the human eyes illusion, given the dithered image.

This problem is ill-posed in the sense of Hadamard [5,6,11]. Using both deterministic and probabilistic alternative methods it is possible to obtain the solution of this problem as the minimum of an energy function [11,13,17,2]. For different ill-posed visual problem many authors use binary elements, called *line variables*, to better recover image-discontinuities [2,13,26,31]. Moreover we inhibit the production of adjacent discontinuities in the image [3,15,16].

The joint minimization of the energy function in the blur mask and the image to be estimated it is a difficult task. In literature to overlay this drawback it is

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 379–388, 2004.
© Springer-Verlag Berlin Heidelberg 2004

often used an iterative technique by both MAP and ML-estimation [8,19,30,31]. An alternative estimation was proposed in [15] in which is determined the blur mask, that used as known in the image restoration, allows to obtain the image that better fits our a priori knowledge of the ideal image. This technique is called IMAP (*Inverse Maximum a Priori*) estimation.

In this paper we propose the use of the IMAP estimation to the problem of inverse dithering. To decrease the computational cost the SA (*Simulated Annealing*) algorithm [2,13,17,18,20,31,32], applied to minimize the energy function, uses an E-GNC (*Extended Graduated Non-Convexity*) algorithm [3] to find the restored image.

2 The Inverse Dithering Problem

Let $\boldsymbol{y}$ be a $n \times n$ black and white image. The inverse dithering problem consists in estimate a $n \times n$ grey-scale levels image $\boldsymbol{x}$ with grey values between 0 (black) and 255 (white).

To this aim is also necessary to estimate a blur operator A such that applied to $\boldsymbol{y}$ returns an image close to $\boldsymbol{x}$. The blur operator A has to simulate the continuous-tone illusion that human eye has believing to see a unique grey-level in a neighborhood of a point. To each operator A it is associated a *blur mask* $M \in \mathbb{R}^{(2h+1)\times(2h+1)}$; from this positive matrix the entries of A can be defined as

$$a_{(i,j),(i+w,j+v)} = \begin{cases} \frac{m_{h+1+w,h+1+v}}{\nu_{i,j}}, & \text{if } |w|, \leq h, \quad |v| \leq h, \\ 0, & \text{otherwise,} \end{cases}$$

where

$$\nu_{i,j} = \sum_{i=\kappa}^{\eta} \sum_{j=\ell}^{\delta} m_{i,j},$$

$$\kappa = \max\{1, 2 + h - i\}, \quad \eta = \min\{2h + 1, n - i + h + 1\},$$

$$\ell = \max\{1, 2 + h - j\}, \quad \delta = \min\{2h + 1, n - j + h + 1\}.$$

The inverse dithering problem is ill-posed in the sense of Hadamard (cf. [5, 6,11]), that is the solution neither exists, nor is unique, nor can be stable in presence of noise. Thus, *regularization techniques* (cf. [2,5,10,11,27,28,29]) are necessary to obtain a stable solution.

To the aim of improving the quality of the results, we introduce some binary elements, called *line variables* [9,13]. These elements have to take in account of the discontinuities present in the grey-levels image $\boldsymbol{x}$, indeed, the ideal images present discontinuities in correspondence with edges of different objects.

We define a *clique* c as the set of points of a square gride on which the first-order finite difference is defined. We indicated with b_c the boolean line variable

associated to clique c; in particular, the one value corresponds to a discontinuity of the involved image in c. The vector $\boldsymbol{b}$ is the set of all line variables b_c. By the notation $c-1$ we denote the clique that precedes c in the order $\preceq$ defined as follows:

$$\{(i,j),(i-1,j)\} \preceq \{(h,j),(h-1,j)\} \quad \Leftrightarrow i \leq h,$$

and

$$\{(i,j),(i,j-1)\} \preceq \{(i,h),(i,h-1)\} \quad \Leftrightarrow j \leq h.$$

We define the solution of inverse dithering problem as the minimum of the following energy function:

$$E(\boldsymbol{x},\boldsymbol{b},A) = \|\boldsymbol{x} - A\boldsymbol{y}\|^2 + \sum_{c\in C}\left[\lambda^2 (D_c\boldsymbol{x})^2(1-b_c) + \alpha b_c\right] + \sum_{c\in C} \varepsilon\, b_c\, b_{c-1}. \quad (1)$$

We refer to this function as *primal energy function.*

The first term of (1) measures faithfulness of the solution to data. The second one is a regularization term, which imposes a smoothness condition on $\boldsymbol{x}$. The *regularization parameter* λ^2*,* allows us to regulate an appropriate degree of smoothing in the solution. The positive parameter α is used in order to avoid to have too many discontinuities in the restored image. The last term is added to the energy function to impose that it should not exist parallel close discontinuities in the restored image. The parameter ε has to be greater or equal to zero.

To the goal to reduce the difficulty of the computation of the minimum of (1) we define the *dual energy function* through the minimization of the primal energy function respect to the line variables b_c. Namely, (cf. [2,7,9,17]):

$$E_d(\boldsymbol{x},A) = \inf_{\boldsymbol{b}} E(\boldsymbol{x},\boldsymbol{b},A).$$

The direct calculation of dual energy function is difficult, so we use the following good approximation of the dual energy function [2,3,15,16]:

$$E_d(\boldsymbol{x},A) = \|\boldsymbol{x} - A\boldsymbol{y}\|^2 + \sum_{c\in C} \psi(D_c(\boldsymbol{x}), D_{c-1}(\boldsymbol{x})), \quad (2)$$

where

$$\psi(t_1,t_2) = \begin{cases} \begin{cases} \lambda^2 t_1^2 & \text{if } |t_1| < s \\ \alpha & \text{if } |t_1| \geq s \end{cases} & \text{if } |t_2| < s \\ \begin{cases} \lambda^2 t_1^2 & \text{if } |t_1| < \bar{s} \\ \alpha + \varepsilon & \text{if } |t_1| \geq \bar{s} \end{cases} & \text{if } |t_2| \geq s. \end{cases} \quad (3)$$

The quantity $s = \sqrt{\alpha}/\lambda$ has the meaning of a *threshold* for creating a discontinuity, while $\bar{s} = \sqrt{\alpha+\varepsilon}/\lambda$ is *suprathreshold* for creating a close parallel discontinuity.

3 IMAP Estimation

In this paper we propose the utilizzation of IMAP (Indirect Maximum a Priori) technique [12,15] to estimate both the blur operator A and the original grey-scale levels image $\boldsymbol{x}$.

Let $\bar{E}_d$ be the term related to the a priori knowledge in the dual energy (2), that is

$$\bar{E}_d(\boldsymbol{x}, A) = \sum_{c \in C} \psi(D_c(\boldsymbol{x}), D_{c-1}(\boldsymbol{x})),$$

where the function ψ is as in (3). Moreover let

$$\boldsymbol{x}(A) = \arg\min_{\boldsymbol{x}} E_d(\boldsymbol{x}, A),$$

that is, the image that minimizes the dual energy when the matrix A is fixed. The IMAP estimation of the solution of the inverse dithering problem can be defined as follows:

$$A = \arg\min_{A} \bar{E}_d(\boldsymbol{x}(A), A), \tag{4}$$

$$\boldsymbol{x} = \boldsymbol{x}(A).$$

In this way, we look for the blur matrix, that used as known in the restoration of a image, allows to obtain the more regular image.

To decrease the computational cost of the algorithm, we use a deterministic algorithm to compute $\boldsymbol{x}(A)$. In fact, deterministic algorithms can give faster estimations of the solution than the ones obtained by stochastic algorithms [7], while in the computation of the minimum in (4) we use the SA (Simulated Annealing) algorithm [2,13,17,18,20,31,32].

4 E-GNC Algorithm

In this section we present an algorithm to minimize the dual energy E_d in the field $\boldsymbol{x}$. In general, E_d is not convex. The algorithms for minimizing a function, when it is not convex, depend on the choice of the starting point. To give an adequate choice of the initial point, a standard technique is to find a finite family of approximating functions $\{E_d^{(p)}\}_p$, such that the first is convex and the last is the original dual energy and then to apply the following algorithm [2,3,4,7,9,14, 16,23]:

initialize $\boldsymbol{x}$*;*
while $E_d^{(p)} \neq E_d$ *do*
find the minimum of the function $E_d^{(p)}$ *starting from the initial point* $\boldsymbol{x}$*;*
$\boldsymbol{x} = \arg\min_{\boldsymbol{x}} E_d^{(p)}$*;*

update the parameter p;
end

Such an algorithm is called GNC (*Graduated Non-Convexity*) algorithm. The first GNC algorithm was proposed by Blake and Zisserman [7,9], who did not consider any constraint on the geometry of the discontinuities. Bedini, Gerace and Tonazzini [3] proposed an extension of GNC algorithm, called E-GNC (*Extended Graduated Non-Convexity*), for the dual energy which takes into account the constraints of non-parallelism.

The Hessian matrix associated with faithfulness to the data in (2), namely to the first term $\|\boldsymbol{x}-A\boldsymbol{y}\|^2$, is the identity matrix, which is positive semidefinite. Thus, in all proposed GNC algorithms, the authors give a family of approximating functions only of the function ψ in (3), while the faithfulness term remains constant in each approximation.

In the E-GNC algorithm the parameter p varies between 1 and 0, and the approximating functions $\psi^{(p)}$ are the following

$$\psi^{(p)}(t_1,t_2)=\begin{cases} g^{(p)}(t_1,\alpha) & \text{if } \mid t_2 \mid \leq s \\ a^{(p)}(t_1)(\mid t_2 \mid -s)^2 + g^{(p)}(t_1,\alpha) & \text{if } s < \mid t_2 \mid \leq \dfrac{u(p)+s}{2} \\ -a^{(p)}(t_1)(\mid t_2 \mid -u(p))^2 + g^{(p)}(t_1,\alpha+\varepsilon) & \text{if } \dfrac{u(p)+s}{2} < \mid t_2 \mid < u(p) \\ g^{(p)}(t_1,\alpha+\varepsilon) & \text{otherwise,} \end{cases}$$

where

$$u(p) = s + pz,$$

with

$$\begin{aligned} z &\geq \sqrt{\frac{4\varepsilon\alpha}{\tau^*}}, \\ z &\geq \frac{2(\sqrt{\alpha}-\sqrt{\alpha(1+\varepsilon)})\sqrt{2\lambda^2+\tau^*}}{\lambda\sqrt{\tau^*}}, \\ z &\geq \frac{8\lambda^2 s}{\sqrt{2\lambda^2\tau^*+\tau^{*2}}}, \\ \tau^* &= \frac{1}{24}, \end{aligned}$$

where

$$g^{(p)}(t,\alpha) = \begin{cases} \lambda^2 t^2 & \text{if} \quad \mid t \mid < q \\ \alpha - (\tau/2)(\mid t \mid -r)^2 & \text{if} \quad q \leq \mid t \mid \leq r, \\ \alpha & \text{if} \quad \mid t \mid > r, \end{cases} \quad \tau = \frac{\tau^*}{p},\ r = \frac{\alpha}{\lambda^2 q},$$

and where finally

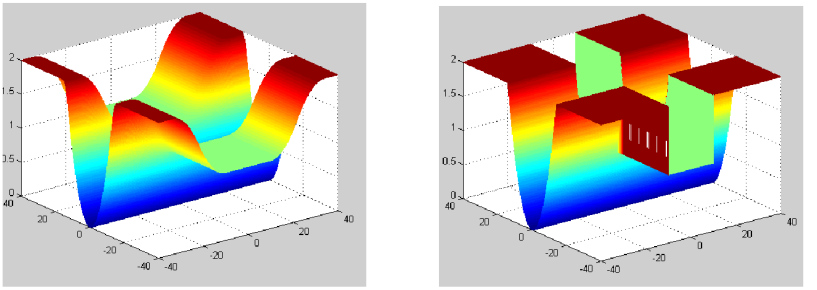


Fig. 1. (a) Function $\psi^{(1)}$; (b) function $\psi^{(0)} \equiv \psi$. They are obtained with $\lambda = 0.1$, $\alpha = 1$ and $\varepsilon = 1$.

Fig. 2. (a) Dithered image; (b) image reconstructed by IMAP estimation with $\lambda = 3$, $\alpha = 2500$ and $\varepsilon = 2500$.

$$a^{(p)}(t) = 2\frac{g^{(p)}(t, \alpha + \varepsilon) - g^{(p)}(t, \alpha)}{[u(p) - s]^2}.$$

In Figures 1(a) and 1(b) the graphs of the functions $\psi^{(1)}$ and $\psi^{(0)} \equiv \psi$, are given respectively.

5 Experimental Results

We have implemented the algorithm of the IMAP estimation in C language on a serial computer. To obtain the dithered image we used the algorithm proposed in [14]. In all our experiments we fixed the values of the free parameter to the following values: $\lambda = 3$, $\alpha = 2500$ and $\varepsilon = 2500$.

(a)

(b)

Fig. 3. (a) Dithered image; (b) image reconstructed by IMAP estimation with $\lambda = 3$, $\alpha = 2500$ and $\varepsilon = 2500$.

(a)

(b)

Fig. 4. (a) Dithered image; (b) image reconstructed by IMAP estimation with $\lambda = 3$, $\alpha = 2500$ and $\varepsilon = 2500$.

In the first experimental result we have considered the 256×256 dithered image shown in Figure 2(a). By IMAP estimation we have obtained the image presented in Figure 2(b), while we have estimated the following blur mask

$$\begin{pmatrix} .82 & 1 & .82 \\ 1 & 1 & 1 \\ .82 & 1 & .82 \end{pmatrix}.$$

The second considered 256×256 dithered image is given in Figure 3(a), while the reconstruction by IMAP is in Figure 3(b). In this case the estimated blur mask is

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}.$$

In Figure 4(a) there is the last considered 128×128 dithered image, and its reconstruction is shown in Figure 4(b). The obtained blur mask is

$$\begin{pmatrix} .62 & 1 & .62 \\ 1 & 1 & 1 \\ .62 & 1 & .62 \end{pmatrix}.$$

6 Conclusions

In this paper, we addressed the inverse dithering problem. Such a problem consists in estimating both the original grey-levels image and the blur mask that best fits the human eyes illusion, given a dithered image.

To solve this problem we have proposed to apply a IMAP estimation. In this way the solution of the problem is defined as the blur mask, that used as known in the image restoration, allows to obtain the image that better fits our a priori knowledge of the ideal image. To decrease the computational costs we have used a E-GNC algorithm for restoring the images during the execution of a SA algorithm. The quality of the experimental results confirms the goodness of such a technique.

References

1. Akarun, L., Yardimci, Y., and Çetin, A. E.: Adaptive Methods for Dithering Color Images. IEEE Trans. Image Process. **6** (1997) 950–955.
2. Bedini, L., Gerace, I., Salerno E., Tonazzini, A.: Models and Algorithms for Edge-Preserving Image Reconstruction. Advances in Imaging and Electron Physics. **97** (1996) 86–189.
3. Bedini, L., Gerace I., Tonazzini, A.: A Deterministic Algorithm for Reconstruction Images with Interacting Discontinuities. CVGIP: Graphical Models Image Process. **56** (1994) 109–123.
4. Bedini, L., Gerace I., Tonazzini, A.: A GNC Algorithm for Constrained Images Reconstruction with Continuous-Valued Line Processes. Pattern Recogn. Lett., **15** (1994) 907–918.
5. Bertero, M., Boccacci, P.: Introduction to Inverse Problems in Imaging. Institute of Physics Publishing, Bristol and Philadelphia. (1998).
6. Bertero, M., Poggio T., Torre, V.: Ill-Posed Problems in Early Vision. IEEE Proc. **76** (1988) 869–889.
7. Blake, A.: Comparison of the Efficiency of Deterministic and Stochastic Algorithms for Visual Reconstruction. IEEE Trans. Pattern Anal. Machine Intell. **11** (1989) 2–12.
8. Bedini, L., Tonazzini,A.: Fast Fully Data-Driven Image Restoration by means of Edge-Preserving Regularization. Real-Time Imaging, Special Issue on Fast Energy–Minimization–Based Imaging and Vision Techniques. **7** (2001) 3–19.
9. Blake, A., Zisserman, A.: Visual Reconstruction. MIT Press, Cambridge, MA. (1987).
10. Charbonnier, P., Blanc-Féraud, L., Aubert, G., Barlaud, M.: Deterministic Edge-Preserving Regularization in Computed Imaging. IEEE Trans. Image Process. **6** (1997) 298–311.

11. Demoment, G.: Image Reconstruction and Restoration: Overview of Common Estimation Structures and Problems. IEEE Trans. Acoust., Speech, and Signal processing. **37** (1989) 2024–2036.
12. Discepoli, M., Gerace, I., Pandolfi, R.: Blind Image Restoration from Multiple Views by IMAP Estimation. To appear in proceeding of Information Processing and Management of Uncertainty in Knowledge-Based Systems IPMU 2004. (2004) pages 8.
13. Geman, S., Geman, D.: Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. IEEE Trans. Pattern Anal. Machine Intell. **6** (1984) 721–740.
14. Gerace, I., Pandolfi, R., Pucci, P.: A new GNC Algorithm for Spatial Dithering. In proceedings of the 2003 International Workshop on Spectral Methods and Multirate Signal Processing SMMSP2003. (2003) 109–114.
15. Gerace, I., Pandolfi, R., Pucci, P.: A new estimation of blur in the blind restoration problem. In proceeding of IEEE International Conference on Image Processing ICIP 2003. (2003) pages 4.
16. Gerace, I., Pucci, P., Boccuto, A., Discepoli, M., Pandolfi, R.: A New Technique for Restoring Blurred Images with Convex First Approximation. In preparation.
17. Geman, D., Reynolds, G., Constrained Restoration and the Recovery of Discontinuities. IEEE Trans. Pattern Anal. Machine Intell. **14** (1992) 367–383.
18. Kirkpatrick, S., Gelatt C. D., Vecchi M.P.: Optimization by Simulated Annealing. Science. **220** (1983) 671–680.
19. Lakshmanan, S., Derin, H.: Simultaneus parameter estimation and segmentation of Gibbs random fields using simulated annealing. IEEE Trans. Pattern Anal. Machine Intell. **11** (1989) 799–813.
20. Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller A. H., Teller E.: Equation of State Calculations by Fast Computing Machines. Journal Chem. Phys. **21** (1953) 1087–1091.
21. Mannos, J. L., Sakrison, D. J: The effects of a visual fidelity criterion on the encoding of image. IEEE Trans. on Information Theory. **20** (1974) 525–536.
22. Mojsilović, A., Solijanin, E.: Color Quantization and Processing by Fibonacci Lattices. IEEE Trans. Image Process. **10** (2001) 1712–1725.
23. Nikolova, M.: Markovian Reconstruction Using a GNC Approach. IEEE Trans. Image Process. **8** (1999) 1204–1220.
24. Özdemir, D., Akarun, L.: Fuzzy Algorithms for Combined Quantization and Dithering. IEEE Trans. Image Process. **10** (2001) 923–931.
25. Puzicha, J., Held, M., Ketterer, J., Buhmann, J. M., Fellner, D. W.: On Spatial Quantization of Color Images. IEEE Trans. Image Process. **9** (2000) 666–682.
26. Raghavan, S., Gupta, N., Kanal, L.: Discontinuity-Preserved Image Flow. In Proceeding of the $11 - th$ International Conference on Pattern Recognition, ICPR92. (1992) 764–767.
27. Teboul, S., Blanc-Féraud, L., Aubert, G., Barlaud, M.: Variational approach for edge-preserving regularization using coupled pde's. IEEE Trans. Image Process. **7** (1998) 387–397.
28. Terzopoulos, D.: Regularization of inverse problems involving discontinuities. IEEE Trans. Pattern Anal. Machine Intell. PAMI-**8** (1986) 413–424.
29. Tikhonov, A. N., Arsenin, V. Y.: Solutions of ill-posed problems. Winston-Wiley, Washington. (1977).
30. Tonazzini, A.: Blur Identification Analysis in Blind Image Deconvolution Using Markov Random Fields. Pattern Recogn. and Image Analysis. **11** (2001) 669–710.

31. Tonazzini, A., Bedini, L.: Degradation Identification and model parameter estimation in discontinuity-adaptive visual reconstruction. Advances in Imaging and Electron Physics. **120** (2002) 193–284.
32. Winkler, G.: Image analysis, random fields and dynamic Monte Carlo methods: a mathematical introduction. Berlin Heidelberg, Springer-Verlag. (2003).

A Study on Neural Networks Using Taylor Series Expansion of Sigmoid Activation Function

Fevzullah Temurtas[1], Ali Gulbag[2], and Nejat Yumusak[1]

[1] Sakarya University, Department of Computer Engineering, Adapazari, Turkey
[2] Sakarya University, Institute of Science & Technology, Adapazari, Turkey

Abstract. The use of microcontroller in neural network realizations is cheaper than those specific neural chips. However, realization of complicated mathematical operations such as sigmoid activation function is difficult via general microcontrollers. On the other hand, it is possible to make approximation to the sigmoid activation function. In this study, Taylor series expansions up to nine terms are used to realize sigmoid activation function. The neural network (NN) structures with Taylor series expansions of sigmoid activation function are used for the concentration estimation of Toluene gas from the trend of the transient sensor responses. The Quartz Crystal Microbalance (QCM) type sensors were used as gas sensors. The appropriateness of the NNs for the gas concentration determination inside the sensor response time is observed with five different terms of Taylor series expansion.

1 Introduction

General hardware implementations of neural networks are the application specific integrated circuits. The application specific neural chips and general purpose ones are more expensive than a microcontroller. One of the most important parts of an artificial neuron is activation function. The sigmoid activation function is perhaps the most popular choice of the activation function. In the hardware implementation concept of neural networks [NNs], it is not so easy to realize complicated mathematical operations such as sigmoid activation functions via microcontrollers. A flexible and software dependent method is required to realize complicated activation functions on microcontrollers. So, adaptation of NNs to the handle systems including microcontrollers for detection of gas concentration is not easy. On the other hand, it is possible to make approximation to the sigmoid activation function and Taylor series expansions can be used to realize the sigmoid activation function [1-3].

Toluene and other volatile organic compounds in ambient air are known to be reactive photo-chemically, and can have harmful effects upon long-term exposure at moderate levels. These type organic compounds are widely used as a solvent in a large number of the chemical industry and in the printing plants [4]. Developing and designing sensors for the specific detection of hazardous components is important [5].

In recent years, a variety of sensitive and selective coating materials have been investigated for chemical sensors. The molecules to be detected interact with the sensitive coating. They may be identified quantitatively by changes of physical or

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 389–397, 2004.
© Springer-Verlag Berlin Heidelberg 2004

chemical parameters of adsorbate/analyte systems such as the refracting index, capacitance, conductivity, total mass, etc. The total mass changes can be monitored by means of quartz crystal microbalance (QCM) sensors, which are widely used as thickness monitors [6,7]. One of the most important problems which is related to the sensor parameters encountered in the high sensitive sensor research is slow response times [8].

Usually, the steady state responses of the sensors are used for concentration estimations of the gases [9-13]. Steady state response means no signals varying in time. But, for realizing the determination of the concentrations before the response times and decreasing the estimation time, the transient responses of the sensors must be used. Transient response means varying signals in time. The response values change very fast at the beginning of the measurement in the transient responses of the sensors. That is, slope is bigger at the beginning and decreases with time. So, an artificial neural network (ANN) structure with tapped time delays is used.

In this study, Taylor series expansions up to nine terms are used to realize sigmoid activation function. These neural network structures with Taylor series expansions of sigmoid activation function and tapped time delays are used for realizing the determination of Toluene gas concentrations from the trend of the transient sensor responses. The performance and the suitability of the method are discussed based on the experimental results.

2 Taylor Series Expansion of Sigmoid Activation Function

The sigmoid activation functions have more implementation area then the existing hard limiting and pure linear activation functions. Equations which used for the sigmoid activation function is given in (1).

$$f(x) = Sigmoid\,(x) = \frac{1}{1+e^{-x}} \tag{1}$$

The main part of the sigmoid activation functions is e-x. Equation which used for Taylor series expansions of e-x [3] is;

$$e^{-x} = 1 - x + \frac{x^2}{2!} - \frac{x^3}{3!} + \ldots + \frac{(-1)^n x^n}{n!} \tag{2}$$

In this study, numbers 3,4,5,6, and 7 are used as n values. That is, Taylor series expansions up to nine terms are used to realize sigmoid activation function. Equation which used for the derivative of the sigmoid activation function in the back propagation [14] algorithm is given in (3),

$$f'(x) = f(x)(1 - f(x)) \tag{3}$$

The generation of sigmoid function makes it possible to design multi layer neural networks on the handle systems including cheap microcontroller.

3 Sensors and Measurement System

The Quartz Crystal Microbalances (QCM) is useful acoustic sensor devices. The principle of the QCM sensors is based on changes Δf in the fundamental oscillation frequency to upon ad/absorption of molecules from the gas phase. To a first approximation the frequency change Δf results from increase in the oscillating mass Δm [15].

$$\Delta f = -\frac{C_f f_0^2}{A}\Delta m \tag{4}$$

where, A is the area of the sensitive layers, C_f the mass sensitivity constant (2.26 10-10 $m^2 s\ g^{-1}$) of the quartz crystal, f_o fundamental resonance of the quartz crystals, Δm mass changes.

The piezoelectric crystals used were AT-Cut, 10 *MHz* quartz crystal (ICM International Crystal Manufacturers Co., Oklahoma, USA) with gold plated electrodes (diameter $\phi = 3$ *mm*) on both sides mounted in a HC6/U holder. The both faces of the two piezoelectric crystals were coated with the phthalocyanine [7]. The instrumentation utilized consist of a Standard Laboratory Oscillator Circuit (ICM Co Oklahoma, USA), power supply and frequency counter (Keithley programmable counter, model 776). The frequency changes of vibrating crystals were monitored directly by frequency counter.

A Calibrated Mass Flow Controller (MFC) (MKS Instruments Inc. USA) was used to control the flow rates of carrier gas and sample gas streams. Sensors were tested by isothermal gas exposure experiments at a constant operating temperature. The gas streams were generated from the cooled bubblers (saturation vapour pressures were calculated using Antoine Equation [16]) with synthetic air as carrier gas and passed through stainless steel tubing in a water bath to adjust the gas temperature. The gas streams were diluted with pure synthetic air to adjust the desired analyte concentration with computer driven MFCs. Typical experiments consisted of repeated exposure to analyte gas and subsequent purging with pure air to reset the baseline. The sensor data were recorded every 2 *s* at a constant of 200 *ml/min*.

In this study, the frequency shifts (*Hz*) versus concentrations (*ppm*) characteristics were measured by using QCM sensor for Toluene (Figure 1). At the beginning of each measurement gas sensor is cleaned by pure synthetic air. Each measurement is composed of six periods. Each period consists of 10 minutes cleaning phase and 10 minutes measuring phase. During the periods of the measurements, at the first period 500 *ppm*, and at the following periods 1000, 3000, 5000, 8000, and 10000 *ppm* gases are given.

4 Neural Network Based Concentration Estimation

A multi-layer feed-forward NN with tapped time delays is used for determination of the concentrations of Toluene from the trend of the transient sensor responses. The network structure is shown in Figure 2. The input, Δf is the sensor frequency shift

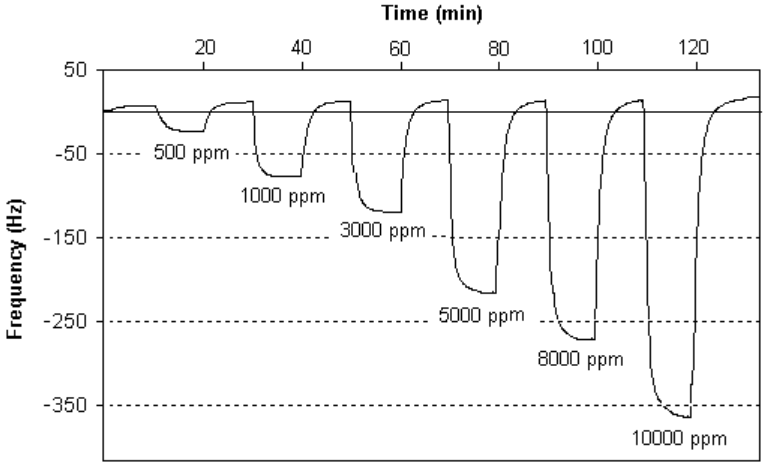


Fig. 1. QCM sensor response for Toluene

value and the output, *PPM* is the estimated concentration. The inputs to the networks are the frequency shift and the past values of the frequency shift. The networks have a single hidden layer and a single output node. The optimum number of hidden layer nodes was determined according to the training results of the NN with sigmoid activation function. This value was used for the NN structures with Taylor series expansions of sigmoid activation function.

Equations used in the neural network model are shown in (5), (6), and (7). As seen from equations, the activation functions for the hidden layer nodes and the output node are sigmoid transfer function. The back propagation algorithm was used for training of neural network model [12].

$$net_j(t) = b_j(t) + \sum_{i=0}^{n} w_{ji}(t)\Delta f(t - i * t_s) \tag{5}$$

$$O_j(t) = f\left(net_j(t)\right) = \frac{1}{1 + e^{-net_j(t)}} \tag{6}$$

$$PPM(t) = \frac{1}{1 + e^{-\left(b(t) + \sum_{j=0}^{m} w_j(t) O_j(t)\right)}} \tag{7}$$

where, $b_j(t)$ are the biases of the hidden layer neurons, $w_{ji}(t)$ are the weights from the input to the hidden layer, $b(t)$ is the bias of the output layer neuron, $w_j(t)$ are the weights from the hidden layer to the output layer, $\Delta f(t - i * t_s)$, i = 1 to n are the past value of the sensor inputs, m is the number of hidden layer nodes and t_s is data sampling time equal to 2 sec.

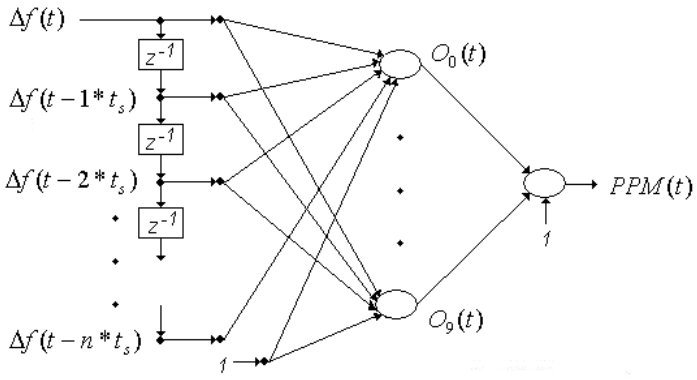


Fig. 2. Multi-layer feed-forward neural network model with a time delayed structure

The information about the trend of the transient sensor responses can be increased by increasing the numbers of data. This requires additional neural network inputs. For illustrating the effect of the numbers of inputs, five different numbers of inputs to the networks are used. These are,

- the frequency shift and the two past values of the frequency shift (three inputs),
- the frequency shift and the four past values of the frequency shift (five inputs),
- the frequency shift and the seven past values of the frequency shift (eight inputs),
- the frequency shift and the nine past values of the frequency shift (ten inputs).

The measured steady state and transient sensors responses were used for the training and test processes.

5 Training of the Networks and Performance Evaluation

The back propagation (BP) method is widely used as a teaching method for an ANN [17]. The BP algorithm with momentum gives the change $\Delta w_{ji}(k)$ in the weight of the connection between neurons i and j at iteration k as;

$$\Delta w_{ji}(k) = -\alpha \frac{\partial E}{\partial w_{ji}(k)} + \mu \Delta w_{ji}(k-1) \tag{8}$$

where, α is the learning coefficient, μ is the momentum coefficient, E is the sum of squared differences error function, and $\Delta w_{ji}(k)$ the weight change in the immediately preceding iteration. Same learning coefficient (0.3) and momentum coefficient (0.7) are used for all of the NN

In this study, the measured steady state and transient sensors responses were used for the training and test processes. Two measurements were made using same QCM sensor for this purpose. One measurement was used as training set and other measurement was used as test set. For the preparation of training and test set, firstly, cleaning phase data removed from the measured responses. Then, the cleaning phase

base frequency shifts (5-15 Hz.) subtracted from sensor responses. Approximately 1800 instantaneous sensor responses were obtained for given six PPM values in both the training and test set. Instantaneous here means values at one point in time. Table 1 shows a replicate data set of Toluene presented to NN structures for the training process.

Table 1. A replicate data set of Toluene presented to NN structures for the training (for NN with 5 inputs)

Normalized NN inputs					Normalized Desired Output $PPM_{true}(t)$
$\Delta f(t)$	$\Delta f(t-t_s)$	$\Delta f(t-2*t_s)$	$\Delta f(t-3*t_s)$	$\Delta f(t-4*t_s)$	
0.000	0.002	0.004	0.006	0.008	0.05
0.002	0.004	0.006	0.008	0.009	0.05
0.004	0.006	0.008	0.009	0.010	0.05
0.006	0.008	0.009	0.010	0.011	0.05
0.008	0.009	0.010	0.011	0.012	0.05
…	…	…	…	…	…
0.023	0.024	0.024	0.024	0.024	0.05
…	…	…	…	…	…
…	…	…	…	…	…
0.000	0.017	0.038	0.076	0.110	1
0.017	0.038	0.076	0.110	0.145	1
0.038	0.076	0.110	0.145	0.173	1
0.076	0.110	0.145	0.173	0.198	1
0.110	0.145	0.173	0.198	0.221	1
…	…	…	…	…	…
0.382	0.383	0.381	0.382	0.382	1

Twenty thousand iterations were used to update weight of the connection between neurons of the ANN structures for all of the training methods.

For the performance evaluation, we have used the mean relative absolute error, E(RAE) [12]:

$$E(RAE) = \frac{1}{n_{test}} \sum_{tetset} \left(\left| \frac{\left(PPM_{predicted} - PPM_{true}\right)}{PPM_{true}} \right| \right) \quad \forall PPM_{true} \neq 0 \tag{9}$$

where, $PPM_{predicted}$ is estimated concentration, PPM_{true} is real concentration and n_{test} is number of test data.

6 Results and Discussions

For determining the optimum number of the hidden layer nodes, five different values of the hidden layer nodes were used. Figure 3 shows the effects of hidden neurons on the performance of the NN with sigmoid activation function. According to Figure 3, the optimum number of hidden layer nodes can be taken as eight. This value was used for the NN structures with Taylor series expansions of sigmoid activation function.

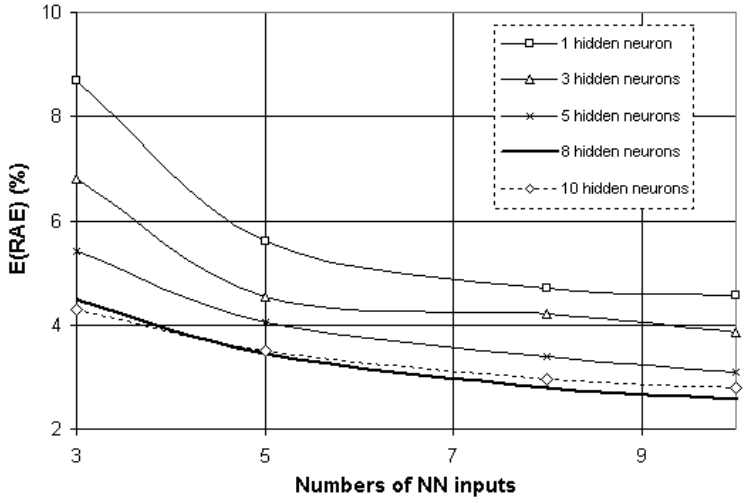


Fig. 3. Error (%) for numbers of hidden neurons versus numbers of ANN inputs graph

Table 2. NN estimation results for Toluene (sensor response time is approximately 250 sec.)

State of responses	# of NN inputs	NN estim. time (sec)	E(RAE) (%)					
			n=3	n=4	n=5	n=6	n=7	Sigm
Transient	3	~ 6	13.23	7.87	6.89	4.87	4.75	4.48
	5	~ 10	9.83	5.46	4.81	3.88	3.65	3.44
	8	~ 16	5.55	3.58	3.47	2.88	2.82	2.78
	10	~ 20	4.47	3.32	3.25	2.73	2.68	2.56

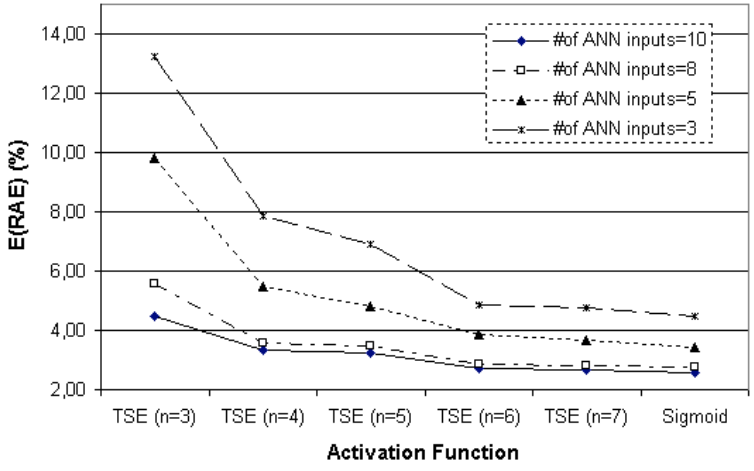


Fig. 4. Error (%) versus activation function

The performance of the NN structures with sigmoid activation function and Taylor series expansions of this function are summarized in Table 2. As seen in this table, acceptable results were obtained for all of the NN structures. From this table, it is also seen that, the NN estimation times are generally at the level of seconds while sensor response times are at the level of minutes. This means that the determination of the concentrations of Toluene from the trend of the transient sensor responses is achieved inside the response time using the NN structures with sigmoid activation function and TSEs of this function.

For easy understanding of the effect of the numbers of the ANN inputs and number of the TSE terms, error (%) versus activation function graph is given in Figure 3. From this figure and above table, it's shown that the increasing number of the TSE terms results improving accuracy at the concentration estimations. From the same table and figure, it's also shown that the increasing number of ANN inputs results improving accuracy at the concentration estimations. And, it can be seen easily that the results of the NNs using TSE of sigmoid activation function with n=6 and 7 (eight and nine terms) are very closer to the results of the NNs using sigmoid activation function. The NNs using TSE of sigmoid activation function with n=3,4, and 5 give also acceptable good results with enough number of NN inputs.

These results mean that, the adaptation of Taylor series expansions to realize sigmoid activation function is achieved and TSE can be used to replace the sigmoid function. The generation of sigmoid function makes it possible to design multi layer neural networks on the handle systems including cheap microcontroller effectively.

In this study, it is seen that acceptable good estimation results can be achieved for the estimation of the Toluene gas concentrations before the steady state response of the QCM sensor using NN structure with enough number of TSE of sigmoid activation function with enough number of NN inputs.

As a result, because of the suitability of the NN structure with TSE of sigmoid activation function, the NN can easily be realized via microcontroller for the handle gas detection systems. And the cost of this realization will be lower than that of the realization via specific neural chips.

References

1. Aybay, I., Çetinkaya, S., ,Halici, U.: Classification of Neural Network Hardware, Neural Network World, IDG Co., Vol. 6, No 1, (1996) 11-29
2. Beiu, V.:, How to Build VLSI-Efficient Neural Chips, Proceedings of the International ICSC Symposium on Engineering of Intelligent Systems, EIS'98, Teneferie, Spain, (1998) 9-13
3. Avci, M., Yildirim, T.: Generation Of Tangent Hyperbolic Sigmoid Function For Microcontroller Based Digital Implementations Of Neural Networks, TAINN 2003, Canakkale,Turkey (2003)
4. Ho, M.H., Gullbault, G.G., Rietz, B.: Continuos Detection of Toluene in Ambient Air with a Coated Piezoelectric Crystal, Anal. Chem., 52(9), (1980)
5. Vaihinger, S., Gopel, W.: Multi - Component Analysis in Chemical Sensing in Sensors: A Comprehensive Survery Ed. W. Gopel, S. Hense, S.N. Zemel, VCH. Weinhe, New York, 2(1) (1991) 192

6. Ozturk, Z.Z., Zhou, R., Wiemar, U., Ahsen, V., Bekaroglu, O., Gopel, W.: Soluble Phthalocyanines For the Detection of Organic Solvents Thin Film Structure With Quartz Microbalance and Capacitance Transducers , Sensors And Actuators B, 26-27 (1995) 208-212
7. Zhou, R., Josse, F., Gopel, W., Ozturk, Z. Z., Bekaroglu, A.: Phthalocyanines As Sensitive Materyals For Chemical Sensors, Applied Organometallic Chemistry, 10 (1996) 557-577
8. Gopel, W., Schierbaum, K.D., Sensors A Comprehensive Survey Ed., chap 1, 18-27, VCH Weinheim, New York, (1991)
9. Szczurek, A., Szecowka, P.M., Licznerski, B.W.: Application of sensor array and neural networks for quantification of organic solvent vapours in air, , Sensors and Actuators B, Vol. 58 (1999) 427-432
10. Temurtas, F., Tasaltin, C., Yumusak, N., Ebeoglu, M.A., Ozturk, Z.Z.: Artificial Neural Networks for the Concentration Estimation of Volatile Organic Gases, Int. Conf. on Elec. and Electronics Eng. (1999), 219-223
11. Pardo, M., Faglia, G., Sberveglieri, G., Corte, M., Masulli, F., Riani, M.: A time delay neural network for estimation of gas concentrations in a mixture, Sensors and Actuators B, 65 (2000) 267–269
12. Temurtas, F., Tasaltin, C., Temurta , H., Yumusak, N., Ozturk, Z.Z.: Fuzzy Logic and Neural Network Applications on the Gas Sensor Data : Concentration Estimation, Lecture Notes in Computer Science, Vol. 2869, (2003), 178-185
13. Caliskan, E., Temurtas, F., Yumusak, N.: Gas Concentration Estimation using Fuzzy Inference Systems, SAU FBE Dergisi, Vol. 8 (1) (2004)
14. Haykin, S.: Neural Networks, A Comprehensive Foundation, Macmillan Publishing Company, Englewood Cliffs, N.J. (1994)
15. King, H. W.: Piezoelectric Sorption Detector, Anal. Chem., 36 (1964) 1735-1739.
16. Riddick, J., Bunger, A., in Weissberger, A., (ed.): Organic Solvents' in Techniques of Chemistry, Volume 2, Wiley Interscience, (1970)
17. Riedmiller, M.: Advanced supervised learning in multilayer perceptrons from backpropagation to adaptive learning algorithms, Int. J. of Computer Standards and Interfaces, Special Issue on Neural Networks, 5 (1994)

A Study on Neural Networks with Tapped Time Delays: Gas Concentration Estimation

Fevzullah Temurtas[1], Cihat Tasaltin[2], Hasan Temurtas[3], Nejat Yumusak[1], and Zafer Ziya Ozturk[2]

[1] Sakarya University, Department of Computer Engineering, Adapazari, Turkey
[2] Tubitak Marmara Research Center, Material and Chem. Tec. Res. Inst., Gebze, Turkey
[3] Dumlupınar University, Department of Electric - Electronic Engineering, Kutahya, Turkey

Abstract. In this study, an artificial neural network (ANN) structure with tapped time delays is used for the concentration estimation of Toluene gas inside the sensor response time by using the transient sensor response. The Quartz Crystal Microbalance (QCM) type sensors were used as gas sensors. A computer controlled measurement and automation system with IEEE 488 card was used to control the gas concentration values and to collect the sensor responses. The determination of Toluene gas concentrations from the trend of the transient sensor responses achieved with acceptable good performances, and the appropriateness of the artificial neural network for the gas concentration determination inside the sensor response time is observed with these training methods.

1 Introduction

Toluene and other volatile organic vapours in ambient air are known to be reactive photo-chemically, and can have harmful effects upon long-term exposure at moderate levels. These type organic compounds are widely used as a solvent in a large number of the chemical industry and in the printing plants [1]. Developing and designing sensors for the specific detection of hazardous components is important [2].

In recent years, a variety of sensitive and selective coating materials have been investigated for chemical sensors. The molecules to be detected interact with the sensitive coating. They may be identified quantitatively by changes of physical or chemical parameters of adsorbate/analyte systems such as the refracting index, capacitance, conductivity, total mass, etc. The total mass changes can be monitored by means of quartz crystal microbalance (QCM) sensors, which are widely used as thickness monitors [3,4].

One of the most important problems which are related to the sensor parameters encountered in the high sensitive sensor research is slow response times [5].

Usually, the steady state responses of the sensors are used for concentration estimations of the gases [6-9]. Steady state response means no signals varying in time. But, for realizing the determination of the concentrations before the response times and decreasing the estimation time, the transient responses of the sensors must be used. Transient response means varying signals in time. The response values change very fast at the beginning of the measurement in the transient responses of the

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 398–405, 2004.
© Springer-Verlag Berlin Heidelberg 2004

sensors. That is, slope is bigger at the beginning and decrease with time. So, an artificial neural network (ANN) structure with tapped time delays is proposed for realizing the determination of Toluene gas concentrations from the trend of the transient sensor responses and decreasing the estimation time. The performance and the suitability of the proposed method are discussed based on the experimental results.

The back propagation (BP) algorithm is widely recognized as a powerful tool for training feed forward neural networks (FNNs). But since it applies the steepest descent method to update the weights, it suffers from a slow convergence rate and often yields suboptimal solutions [10,11]. A variety of related algorithms have been introduced to address that problem. A number of researchers have carried out comparative studies of MLP training algorithms [12-15]. The BP with momentum and adaptive learning rate algorithm [12], Resilient BP [12], Fletcher-Reeves conjugate gradient algorithm [12,13], Broyden, Fletcher, Goldfarb, and Shanno quasi-Newton algorithm [12,14], and Levenberg-Marquardt algorithm [12,15] used in this study are these type algorithms.

2 Sensors and Measurement System

The Quartz Crystal Microbalances (QCM) is useful acoustic sensor devices. The principle of the QCM sensors is based on changes Δf in the fundamental oscillation frequency to upon ad/absorption of molecules from the gas phase. To a first approximation the frequency change Δf results from increase in the oscillating mass Δm [16].

$$\Delta f = -\frac{C_f f_0^2}{A}\Delta m \tag{1}$$

where, A is the area of the sensitive layers, C_f the mass sensitivity constant (2.26 10-10 $m^2 s\ g^{-1}$) of the quartz crystal, f_o fundamental resonance of the quartz crystals, Δm mass changes.

The piezoelectric crystals used were AT-Cut, 10 MHz quartz crystal (ICM International Crystal Manufacturers Co., Oklahoma, USA) with gold plated electrodes (diameter $\phi = 3\ mm$) on both sides mounted in a HC6/U holder. The both faces of the two piezoelectric crystals were coated with the phthalocyanine [4]. The instrumentation utilized consist of a Standard Laboratory Oscillator Circuit (ICM Co Oklahoma, USA), power supply and frequency counter (Keithley programmable counter, model 776). The frequency changes of vibrating crystals were monitored directly by frequency counter.

A Calibrated Mass Flow Controller (MFC) (MKS Instruments Inc. USA) was used to control the flow rates of carrier gas and sample gas streams. Sensors were tested by isothermal gas exposure experiments at a constant operating temperature. The gas streams were generated from the cooled bubblers (saturation vapour pressures were calculated using Antoine Equation [17]) with synthetic air as carrier gas and passed through stainless steel tubing in a water bath to adjust the gas temperature. The gas streams were diluted with pure synthetic air to adjust the desired analyte concentration with computer driven MFCs. Typical experiments consisted of repeated

exposure to analyte gas and subsequent purging with pure air to reset the baseline. The sensor data were recorded every 2 *s* at a constant of 200 *ml/min*.

In this study, the frequency shifts (*Hz*) versus concentrations (*ppm*) characteristics were measured by using QCM sensor for Toluene (Figure 1). At the beginning of each measurement gas sensor is cleaned by pure synthetic air. Each measurement is composed of six periods. Each period consists of 10 minutes cleaning phase and 10 minutes measuring phase. During the periods of the measurements, at the first period 500 *ppm*, and at the following periods 1000, 3000, 5000, 8000, and 10000 *ppm* gases are given.

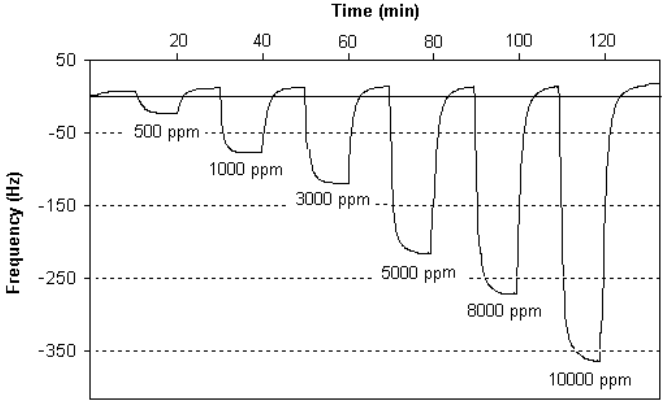


Fig. 1. QCM sensor response for Toluene

3 Neural Network Based Concentration Estimation

A multi-layer feed-forward ANN with tapped time delays is used for determination of the concentrations of Toluene from the trend of the transient sensor responses. The network structure is shown in Figure 2. The input, Δf is the sensor frequency shift value and the output, *PPM* is the estimated concentration. The inputs to the networks are the frequency shift and the past values of the frequency shift. The networks have a single hidden layer and a single output node. The optimum number of hidden layer nodes was determined according to the training results of the back propagation with momentum and adaptive learning rate algorithm. This value was used for other training algorithms.

Equations used in the neural network model are shown in (2), (3), and (4). As seen from equations, the activation functions for the hidden layer nodes and the output node are tangent-sigmoid transfer function. The back propagation algorithm was used for training of neural network model [8].

$$net_j(t) = b_j(t) + \sum_{i=0}^{n} w_{ji}(t)\Delta f(t - i * t_s) \tag{2}$$

$$O_j(t) = f\left(net_j(t)\right) = \frac{1}{1+e^{-net_j(t)}} \tag{3}$$

$$PPM(t) = \frac{1}{1+e^{-\left(b(t)+\sum_{j=0}^{m} w_j(t)O_j(t)\right)}} \tag{4}$$

where, $b_j(t)$ are the biases of the hidden layer neurons, $w_{ji}(t)$ are the weights from the input to the hidden layer, $b(t)$ is the bias of the output layer neuron, $w_j(t)$ are the weights from the hidden layer to the output layer, $\Delta f(t-i*t_s)$, i = 1 to n are the past value of the sensor inputs, m is the number of hidden layer nodes and ts is data sampling time equal to 2 sec.

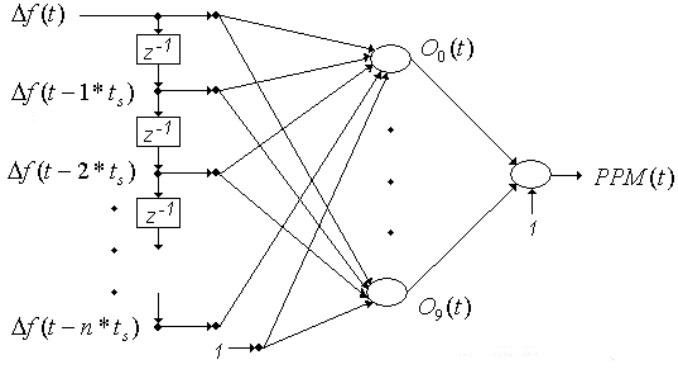


Fig. 2. Multi-layer feed-forward neural network model with a time delayed structure

The information about the trend of the transient sensor responses can be increased by increasing the numbers of data. This requires additional neural network inputs. For illustrating the effect of the numbers of inputs, five different numbers of inputs to the networks are used. These are,

- the frequency shift (one input),
- the frequency shift and the two past values of the frequency shift (three inputs),
- the frequency shift and the four past values of the frequency shift (five inputs),
- the frequency shift and the seven past values of the frequency shift (eight inputs),
- the frequency shift and the nine past values of the frequency shift (ten inputs).

Steady state responses of the sensor were also used in the concentration estimation for the comparison. For this purpose an ANN which has a single input node, a single hidden layer with ten hidden layer nodes and a single output node were used.

4 Training of the Networks and Performance Evaluation

The back propagation (BP) method is widely used as a teaching method for an ANN. The main advantage of the BP method is that the teaching performance is highly improved by the introduction of a hidden layer [12]. In this paper, five different type high performance BP training algorithms which use different optimization techniques were used. These are, BP with momentum and adaptive learning rate (GDX) [12], Resilient BP (RP) [12], Fletcher-Reeves conjugate gradient algorithm (CGF) [12,13], Broyden, Fletcher, Goldfarb, and Shanno quasi-Newton algorithm (BFG) [12,14], and Levenberg-Marquardt algorithm (LM) [12,15]. Detailed information about these training algorithms can be found in [12-15].

In this study, the measured steady state and transient sensors responses were used for the training and test processes. Two measurements were made using same QCM sensor for this purpose. One measurement was used as training set and other measurement was used as test set. For the preparation of training and test set, firstly, cleaning phase data removed from the measured responses. Then, the cleaning phase base frequency shifts (5-15 Hz.) subtracted from sensor responses. Approximately 1800 instantaneous sensor responses were obtained for given six PPM values in both the training and test set. Instantaneous here means values at one point in time.

One thousand iterations were used to update weight of the connection between neurons of the ANN structures for all of the training methods.

For the performance evaluation, we have used the mean relative absolute error, E(RAE) [8]:

$$E(RAE) = \frac{1}{n_{test}} \sum_{tetset} \left(\left| \frac{\left(PPM_{predicted} - PPM_{true} \right)}{PPM_{true}} \right| \right) \quad \forall PPM_{true} \neq 0 \tag{5}$$

where, $PPM_{predicted}$ is estimated concentration, PPM_{true} is real concentration and n_{test} is number of test data.

5 Results and Discussions

For determining the optimum number of the hidden layer nodes, five different values of the hidden layer nodes were used. Figure 3 shows the effects of hidden neurons on the performance of the ANN for the GDX algorithm. According to Figure 3, the optimum number of hidden layer nodes can be taken as eight. This value was used for the RP, CGF, BFG, and LM training algorithms.

One thousand iterations were used for all of the training methods. Figure 4 gives the error *E(RAE)* (%) for the training algorithms versus numbers of iterations graph for Toluene gas. To show the detailed difference of the training algorithm, logarithmic axis for *E(RAE)* (%) was used. The training results of the RP and BFG algorithms are almost similar for this study as seen in Figure 4. The results for the CGF algorithm are also closer to those of them. These three algorithms seem a bit faster than GDX algorithm for this study. From the same figure, it can be seen easily

that LM training algorithm provides faster convergence than other algorithms in the concentration estimation of Toluene for this study.

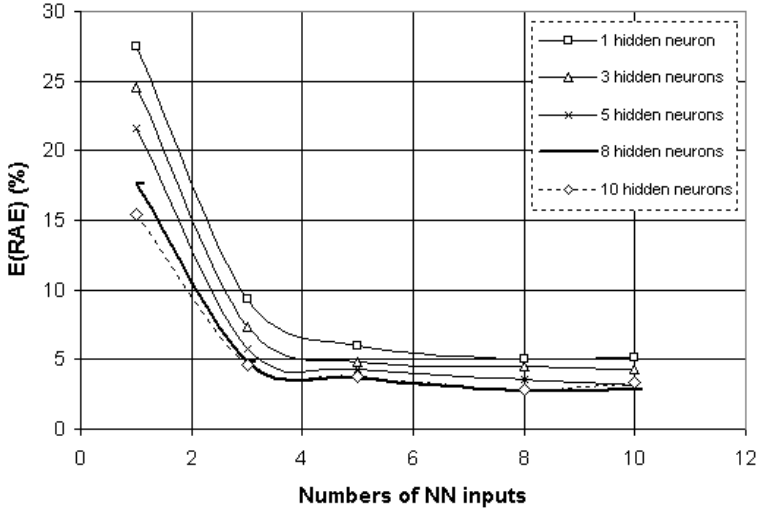


Fig. 3. Error (%) for numbers of hidden neurons versus numbers of ANN inputs graph for Toluene for 1000 iterations (GDX)

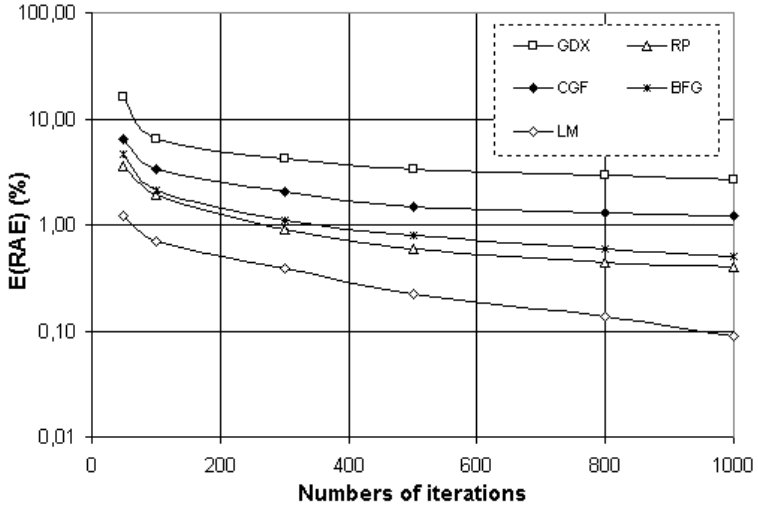


Fig. 4. Error (%) for training algorithms versus numbers of iterations for Toluene

The ability of the ANN to estimation of Toluene concentrations related to number of ANN inputs are summarized in Table 1 for the training methods. As seen in this table, acceptable good results were obtained for the all training algorithm. From this

table, it is also seen that, ANN estimation times are generally at the level of seconds while sensor response times are at the level of minutes. This means that the determination of the concentrations of Toluene from the trend of the transient sensor responses is achieved inside the response time using ANN with tapped time delays with the GDX, RP, CGF, BFG, and LM training methods.

Table 1. ANN estimation results for Toluene (Sensor response time is approximately 250 sec.)

State of responses	# of ANN inputs	ANN estimation time (sec)	E(RAE) (%)				
			GDX	RP	CGF	BFG	LM
Transient	1	2	17.7	11.9	13.9	13.2	8.3
	3	6	5.7	2.2	2.9	2.2	0.8
	5	10	4.5	0.9	1.8	1.1	0.3
	8	16	3.8	0.5	1.4	0.7	0.1
	10	20	2.7	0.4	1.2	0.5	0.1
Steady state	1	~250	2.3	0.2	0.7	0.2	0.0

For easy understanding of the effect of the numbers of the ANN inputs, error *E(RAE)* (%) versus numbers of ANN inputs graph is given in Figure 5. From this figure and above table, it's shown that the increasing number of ANN inputs results improving accuracy at the concentration estimations. This can be because of the fact that the information about the trend of the transient sensor responses can be increased by increasing the numbers of data and this requires additional neural network inputs. When the numbers of ANN inputs are 8 and 10, the estimation results from transient responses are very closer to estimation results from steady state responses for one thousand iterations especially with LM training algorithm.

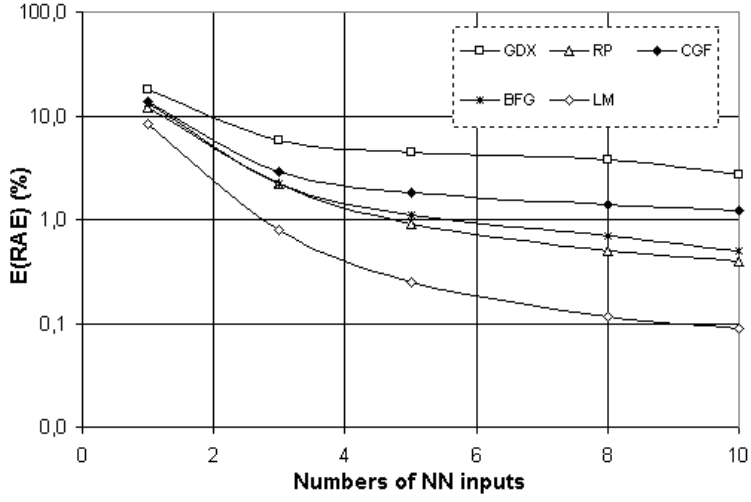


Fig. 5. Error (%) versus numbers of ANN inputs graph for Toluene

In this study we saw that the proposed ANN structure with tapped time delays is useful for realizing the determination of the concentrations inside the response times and optimum estimation results can be achieved by using enough number of ANN inputs and suitable training algorithm.

References

1. Ho, M.H., Gullbault, G.G., Rietz, B.: Continuos Detection of Toluene in Ambient Air with a Coated Piezoelectric Crystal, Anal. Chem., 52(9), (1980)
2. Vaihinger, S., Gopel, W.: Multi - Component Analysis in Chemical Sensing in Sensors: A Comprehensive Survery Ed. W. Gopel, S. Hense, S.N. Zemel, VCH. Weinhe, New York, 2(1) (1991) 192
3. Ozturk, Z.Z., Zhou, R., Wiemar, U., Ahsen, V., Bekaroglu, O., Gopel, W.: Soluble Phthalocyanines for the Detection of Organic Solvents Thin Film Structure with Quartz Microbalance and Capacitance Transducers, Sensors and Actuators B, 26-27 (1995) 208-212
4. Zhou, R., Josse, F., Gopel, W., Ozturk, Z. Z., Bekaroglu, A.: Phthalocyanines as Sensitive Materyals For Chemical Sensors, Applied Organometallic Chemistry, 10 (1996) 557-577
5. Gopel, W., Schierbaum, K.D.: Sensors A Comprehensive Survey Ed., chap 1, 18-27, VCH Weinheim, New York, (1991)
6. Szczurek, A., Szecowka, P.M., Licznerski, B.W.: Application of sensor array and neural networks for quantification of organic solvent vapours in air, Sensors and Actuators B, Vol. 58 (1999) 427-432
7. Pardo, M., Faglia, G., Sberveglieri, G., Corte, M., Masulli, F., Riani, M.: A time delay neural network for estimation of gas concentrations in a mixture, Sensors and Actuators B, 65 (2000) 267–269
8. Temurtas, F., Tasaltin, C., Temurta , H., Yumusak, N., Ozturk, Z.Z.: Fuzzy Logic and Neural Network Applications on the Gas Sensor Data: Concentration Estimation, Lecture Notes in Computer Science, Vol. 2869, (2003), 178-185
9. Caliskan, E., Temurtas, F., Yumusak, N.: Gas Concentration Estimation using Fuzzy Inference Systems, SAU FBE Dergisi, Vol. 8 (1) (2004)
10. Gori, M., Tesi, A.: On the problem of local minima in backpropagation, IEEE Trans. Pattern Analysis and Machine Intelligence, 14, (1992) 76–85
11. Brent, R.P.: Fast Training Algorithms for Multi-layer Neural Nets, IEEE Transactions on Neural Networks 2 (1991) 346–354
12. Hagan, M. T., Demuth, H. B., Beale, M. H.: Neural Network Design, Boston, MA: PWS Publishing, (1996)
13. Fletcher, R., Reeves, C. M.: Function minimization by conjugate gradients, Computer Journal, vol. 7, (1964) 149-154
14. Dennis, J. E., Schnabel, R.B.: Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Englewood Cliffs, NJ: Prentice-Hall, (1983).
15. Hagan, M. T., Menhaj, M.: Training feedforward networks with the Marquardt algorithm, IEEE Transactions on Neural Networks, vol. 5 (6), (1994) 989-993.
16. King, H. W.: Piezoelectric Sorption Detector, Anal. Chem., 36 (1964) 1735-1739.
17. Riddick, J., Bunger, A., in Weissberger, A., (ed.): Organic Solvents' in Techniques of Chemistry, Volume 2, Wiley Interscience, (1970)

Speech Emotion Recognition and Intensity Estimation

Mingli Song, Chun Chen, Jiajun Bu, and Mingyu You

College of Computer Science, Zhejiang University, Hangzhou 310027, P.R.China

Abstract. In this paper, a system for speech emotion analysis is presented. On a corpus of over 1700 utterances from an individual, the feature vector stream is extracted for each utterance based on short time log frequency power coefficients (LFCC). Using the feature vector streams, we trained Hidden Markov Models (HMMs) to recognize seven basic categories emotions: neutral, happiness, anger, sadness, surprise, fear. Furthermore, the intensity of the basic emotion is divided into 3 levels. And we trained 18 sub-HMMs to identify the intensity of the recognized emotions. Experiment result shows that the emotion recognition rate and the estimation of intensity performed by our system are of good and convincing quality.

1 Introduction

Emotion recognition is formed to be an active research area in these years. As we know, emotion recognition can be used in the human/computer communication, speech-driven facial animation, etc. Therefore, a lot of researches have been undertaken to perform emotion recognition based on different media.

As we know, though speaking the same content, people may have different expressions because of their different emotions. Theoretically, the emotion can be estimated from the speech signals according to predefined feature parameters[1, 2,3,4]. Furthermore, an SVM-based emotion recognition method was proposed in [5] so as to obtain speech-driven cartoon animation and the result is worth of appreciation. However, it is known that people may have the emotions at different levels even if they are similar according to the basic emotion definition[6]. Namely, The intensity of emotion wasn't considered in past approaches even though the specific emotion was actually at different levels actually.

For an intelligent human/machine system, it is very important that determine not only the basic emotion style but also the intensity of it. Lien[7] proposed a method to extract expression intensity through calculation of the sum-of-squared-difference (SSD) of the high gradient components in frames. However, the computation of the high gradient components of each image is complicated and not flexible for general application.

In our work, firstly, we build a robust system to analyze the acoustic signals to extract the input feature vector streams based on log frequency power coefficient(LFPC)[8]. Secondly, seven HMMs(Hidden Markov Models) are

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 406–413, 2004.
© Springer-Verlag Berlin Heidelberg 2004

trained to classify the basic emotions . Secondly, an emotion intensity estimation method is proposed to recognize the emotion more precisely.

The remainder of the paper is organized as follows. Section 2 outlines the framework of our system. Section 3 describes the speech features and the recognition model in our approach. Section 4 gives out the emotion intensity estimation method based on the emotion vector analysis. In section 5, the experiment result is provided and analyzed. Finally, we conclude in section 6 with discussion and future work.

2 System Overview

Fig.1 shows an overview of our system. Our system consists of two parts: basic emotion recognition and emotion intensity estimation. First, a basic speech emotion recognition system is developed. To train the emotion recognizer, we collected a corpus containing over 1700 utterances from one speaker with variety of emotions. Hidden Markov Models(HMMs) are trained to classify these utterances into seven categories: neutral, anger, sadness, surprise, disgust, joy and fear.

The emotion intensity estimation is performed by clustering the speech eigenvectors into three levels using sub-HMMs. To train these sub-HMMs, the sample utterances are labelled with their basic emotion and intensity level in advance. And these sub-HMMs are applied immediately after the system determine what basic emotion style the input utterance should be.

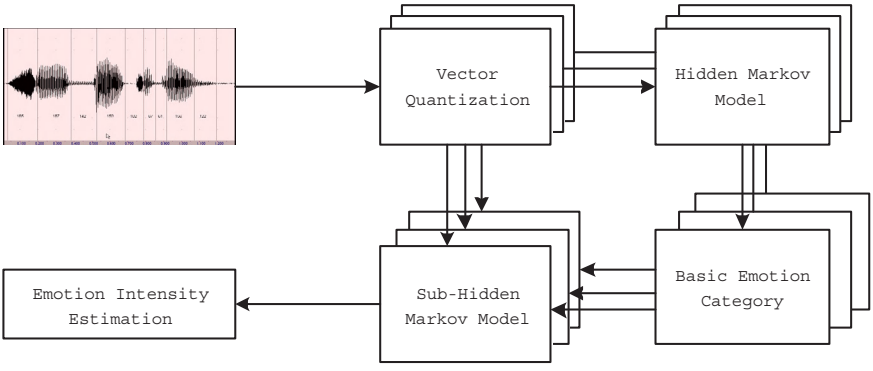


Fig. 1. System overview

3 Basic Emotion Recognition

As an important human behavior for conveying psychological information, emotion has been studied for centuries. In human-computer interaction, the computer can be made to produce more intelligent responses if the state of human

emotion person can be accurately identified. There are two broad types of information in speech: semantic and acoustic. The semantic part of the speech carries linguistic information while it does not always reflect the state of emotion. So much work has been done for emotion analysis in the acoustics field. It was showed that a high correlation between some statistical measures of speech and the emotional state of the speaker[9,10]. For instance, anger usually implies a high values of pitch and fast speaking rates, while sadness have been associated with lower standard deviation of pitch and slower speech rates.

3.1 Selection of Feature Vectors for Emotion Modelling

Similar to several other approaches of recognizing emotion from speech, we classify emotions into seven basic but representative ones: neutral, joy, anger, disgust,surprise, sadness and fear. To make the recognition more accurately, the choice of the acoustic features is very important.

For speech recognition, Linear prediction Cepstral coefficients(LPCC), mel-frequency Cepstral coefficients(MFCC) and short time log frequency power coefficients(LFPC) are the popular choices as features representing the phonetic content of speech. And, it was found that the intensity, the spectral distribution, the speaking rate, and the fundamental frequency contour are important features that discriminate the emotion states. The choice of features takes important role in these processes. Tin Lay Nwe[8] analyzed and compared these three coefficients and concluded that LFPC performed best with the average accuracy of 77.1%. So LFPC is better for recognition of speech emotion compared with LPCC and MFCC that are used for speech recognition.

In our approach, LFPC are used as the emotion features, which are DFT-based and can be extracted with filter in different log frequency bands from 200Hz to 3.9kHz. A log frequency filter bank is a model that follows the varying auditory resolving power of the human ear for various frequencies. The filter bank divides the speech signal into 12 frequency bands that match the critical perceptual bands of the human ear. The sample of each frame is windowed by $W_m(k)$[11]

$$W_m(k) = \begin{cases} 1, l_m \le k \le h_m \\ 0, \quad otherwise \end{cases}, m = 1, 2, \ldots, 12 \tag{1}$$

where k is the DFT domain index, l_m and h_m are the lower and upper edges of mth filter bank. So the mth filter bank output is derived[12]:

$$S_t(m) = \sum_{k=f_m-(b_m/2)}^{f_m-(b_m/2)} (X_t W_m(k))^2, m = 1, 2, \cdots, 12 \tag{2}$$

where $X_t(k)$ is the kth spectral component of the windowed signal, t is the frame number, $S_t(m)$ is the output of the mth filter bank, and f_m, b_m are the center frequency and the bandwidth of the mth sub-band respectively.

$SE_t(m)$ below is the parameter that indicates the energy distribution among sub-bands.

$$SE_t(m) = \frac{10 \lg(S_t(m))}{N_m} \tag{3}$$

N_m is the number of spectral components in the mth filter bank. Consequently, 12 LFPCs are obtained for each speech frame.

3.2 Data Preparation

After feature parameter extraction, a 12 dimensional vector is obtained corresponding to each frame of the speech sample. Before compressing the data, all the extracted coefficients are normalized. The specific preprocessing is vector quantization. K-means algorithm is applied to cluster the vectors into 64 groups to form the codebook. All vectors falling into a particular cluster are coded with the vector representing the cluster.

A vector of 12 short time LFPC representing each frame is assigned to a cluster by vector quantization. With the frames from 1 to n, if the codebook cluster is m_c, then the vector f_n is assigned the codeword $v_n^{'}$ according to the following equation.

$$v_n^{'} = \arg \min_{1 \le v \le V} d(f_n, m_c) \tag{4}$$

For a speech utterance with T frames, a feature vector E is

$$E = (v_1^{'}, v_2^{'}, \cdots, v_T^{'}) \tag{5}$$

3.3 Speech Emotion Recognition

Hidden Markov Model(HMM) is a good method to process temporal-space signal and hence it is widely used in speech recognition. In our system, an ergodic model HMM is adopted to classify the emotions implied in speech. Every state in ergodic model HMM is reachable in a single step from every other state. Experimentally, the number of the states in the model is suitable to be 4. To perform as a recognizer, the model should be trained first. Maximum likelihood training is a well understood technique. However, the iterative maximum likelihood estimation of the parameters only converges to a local optimum, making the choice of the initial parameters of the model a critical issue. The parameters of the HMM used in our system are defined below:

Symbol Sequence:	$E = (v_1^{'}, v_2^{'}, \cdots, v_T^{'})$
States:	$q = (q_1 = 1, \cdots, q_t = i,)$
Codebook Size:	64
HMM Parameter Set:	$\lambda = (\pi, A, B)$
Initial State Distribution:	$\pi_1 = 1.0, \quad \pi_k = 0.0 \quad if \quad 2 \le k \le N$
State-transition Probability:	$A_{N \times N} = \{a_{ij}\}$
Observable Symbol Probability:	$B_{M \times N} = \{b_j(o_{t+1})\}$ *at state j and time* t+1
Output Probability:	$P(O\|\lambda)$

Topology of the HMM is shown in Figure 2.

Different from the Forward algorithm used in [8] for training the models. we employ Viterbi algorithm in our approach, which is proved to be a good method to determine the optimal sequence of states for the nodes of the speech

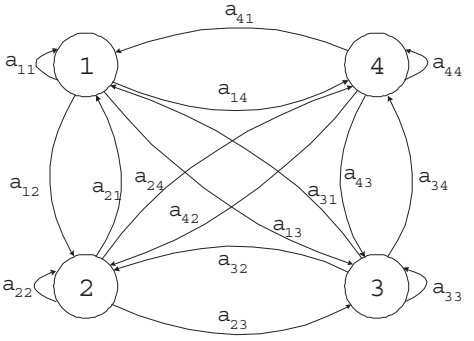


Fig. 2. The Topology of Hidden Markov Model

vector stream that maximize the observation likelihood[13,14]. The basic idea of the Viterbi algorithm is similar to the Forward procedure whose calculation at each time is considered only between two consecutive times t and $t+1$, and starts at the initial time $t = 1$ and proceeds forward to the end time $t = T$. The major difference exists during this calculation between two instant times. The control, which produces the maximum value corresponding to the single shortest or best path(state sequence), is temporarily saved instead of summing of overall calculations. And, at the end of the state sequence for the calculation, the "saved" best controls can be used to recover the state space trajectory based on path backtracking. Detail of the training and re-estimation algorithms is given in [14]. After training, seven HMMs are established for the speaker, one for each class.

4 Emotion Intensity Estimation

Previous approaches didn't consider the estimation of emotion intensity from the speech. In our system, the training utterances for each basic emotion are further segmented into 3 levels manually in advance: low, middle and high. We train sub-HMMs of each level with these utterance vector streams. Finally, 3 models are trained for every basic emotion style except the neutral one because it is known that the neutral emotion has no intensity information. So totally there are 18 sub-HMMs trained for all the 7 basic emotion styles. The training and re-estimation algorithm of sub-HMMs is same to the basic one mentioned in section 3. To reduce the calculating complexity and make the recognition more precise, these sub-HMMs aren't applied directly to the input speech vector stream. Actually, The intensity estimation will not be carried out until the basic emotion of the utterance is identified.

Table 1. Distribution of Training Utterances

	Basic Emotions						
Level	**Neutral**	**Happiness**	**Surprise**	**Fear**	**Anger**	**Sadness**	**Disgust**
Low		60	80	75	60	76	67
Middle		50	70	50	50	32	54
High		50	40	35	47	35	43
Total	100	160	190	160	157	143	164

Table 2. Result of basic emotion recognition and intensity estimation. Columns represents the emotion elected in first choice for utterances belonging to the emotion of each row, where N stands for neutral, H for Happiness, S for sadness, F for fear, A for anger, T for surprise, D for disgust.

		Basic Emotions							
		N	**H**	**S**	**F**	**A**	**T**	**D**	**Total**
Neutral		67	0	0	0	0	0	0	67
Happiness	Low	6	34	3	1	1	0	0	105
	Middle	2	50	3	0	0	0	0	
	High	0	5	0	0	0	0	0	
Sadness	Low	9	0	45	0	0	0	0	85
	Middle	0	0	30	0	0	0	0	
	High	0	0	1	0	0	0	0	
Fear	Low	1	0	2	45	0	2	1	102
	Middle	0	0	0	50	0	0	0	
	High	0	0	0	1	0	0	0	
Anger	Low	0	1	0	4	52	0	2	105
	Middle	0	0	0	1	43	0	1	
	High	0	0	0	0	1	0	0	
Surprise	Low	0	4	0	5	3	68	2	147
	Middle	0	1	0	1	1	56	0	
	High	0	0	0	1	0	5	0	
Disgust	Low	1	2	1	1	2	0	56	90
	Middle	0	0	0	0	0	1	23	
	High	0	0	0	0	0	0	3	
Total		86	97	85	110	103	132	88	701

5 Experiment and Result

Experiments are conducted to evaluate the performance of the proposed system. We built a corpus which consists of over 1700 utterances. These utterances are marked with two labels: the basic emotion type and the intensity level except the neutral ones. Table 1 shows the distribution of the sample utterances in the database. The total number of utterances for each basic emotion are used to train the basic emotion recognition HMMs, while the utterances at different levels are used to train the sub-HMMs. Table 2 presents the percentage of correctly

identified emotions and their intensity for the 701 utterances. The recognition rate of basic emotion is higher than 84%, which is appreciable with the best accuracy of 94.1%. Furthermore, the intensity of the basic emotion is classified into 3 levels automatically with the sub-HMMs.

6 Conclusion

In this paper, an approach for analyzing of emotional state of utterance is proposed. The system makes use of short time LFPC for feature representation. And 7 viterbi algorithm-based ergodic HMMs are employed to classify the utterances into 7 basic emotion styles, then 18 sub-HMMs are applied to identify the intensity of the recognized basic emotions. While the result is encouraging, there remain a number of areas to be further explored. First, the emotion recognition can be improved. And expressive facial animation can be performed based on this approach. Currently, we plan to integrate visual analysis into our system to enhance the accuracy of recognition.

Acknowledgements. This work is partly supported by NSFC grants 60203013 and HP laboratory of Zhejiang University.

References

1. L.R. Rabiner, B.H. Juan: Fundamentals of Speech Recognition, Prentice Hall, 1993
2. F.Dellaert, T.Polzin and A.Waibel: Recognizing emotion in speech. Proceedings of ICSLP 1996.
3. R.Nakatsu, J.Nicholson, and N.Tosa: Emotion recognition and its application to computer agents with spontaneous interactive capabilityies. Proceedings of the third conference on Creativity and Cognition, 1999, page(s): 135-143
4. A.Paeschke, W.F.Sendlmeirer: Prosodic characteristics of emotional speech: Measurements of fundamental frequency movements. Proceedings of ISCA-Workshop on Speech and Emotion, 2000
5. Yan Li, Feng Yu, Ying-Qing Xu, Eric Cheng, Heung-Yeung Shum: Speech-driven cartoon animation with emotions. Proceedings of the ninth ACM international conference on Multimedia, 2001
6. P. Ekman, W.V. Friesen: Facial Action Coding System: Investigator's Guide, Consulting Psychologists Press, 1978.
7. James J.J. Lien, Takeo Kanade, Jeffrery F. Cohn, Ching-Chung Li:Subtly different facial expression recognition and expression intensity estimation. Proceedings of IEEE Conference on Computer Vison and Pattern Recogntion, June, 1998, page(s): 853-859
8. Nwe, Tin Lay, Foo, Say Wei, De Silva, Liyanage C.:Speech emotion recognition using hidden Markov models,Speech Communication Volume: 41, Issue: 4, November, 2003, page(s): 603-623
9. N.Amir,S.Ron:Towards an automatic classicification of emotion in speech, Proceedings of ICSLP'98, December, 1998, page(s): 225-228
10. V. Petrushin:Emotion recognition in speech signals: Experimental study, development, and application, Proceedings of the ICSLP 2000, June, 2000.

11. Becchetti, C., Ricotti, L.P.:Speech Recognition Theory and C++ Implemetnation, John Willey & Sons, New York, 1998
12. Nwe, Tin Lay; Foo, Say Wei; De Silva, Liyanage C.:Classification of stress in speech using linear and nonlinear features, Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003, Volume: 2, page(s): 9-12
13. Forney G.D.:The Viterbi Algorithm, Proceedings of the IEEE. Vol.61, No.3, March 1973, page(s): 268-278
14. Rabiner L.R.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, Proceedings of The IEEE. Vol. 77, No.2, February 1989, page(s): 257-285

Speech Hiding Based on Auditory Wavelet

Liran Shen, Xueyao Li, Huiqiang Wang, and Rubo Zhang

College Of Computer Science And Technology, Harbin Engineering University, Harbin 150001, People's Republic of China
shenliran@0451.com, yinq2003@eyou.com

Abstract. A novel method to embed secret speech into open speech is proposed. The secret speech is coded into binary parameter bits with Mix-Excitation Linear Prediction (MELP) algorithm, and the bits are used to form hiding information sequence. The open speech is automatically divided into voiced frames and unvoiced frame using auditory wavelet transform. One voice frame, the auditory wavelet transform was used to detect pitch, and the pitch is utilized to the current embedding position in open speech. The information hiding procedure is completed by modifying relevant wavelet coefficients. At the receiver, based on the same pitch detection method, the embedding position is found and the hiding bit is recovered. The secret speech can be received after MELP decoding. The experiments show that the method is strongly robust to many attacks such as compression, filter and so on.

1 Introduction

The data hiding or digital watermarking technique has been required and developed for the copyright protection and authentication of multimedia contents. In the evaluation of the data hiding performance, not only imperceptibility of watermarks but also the robustness against signal processing such as the AD/DA conversion, the encoding and decoding of source signals is required. Among the data hiding techniques for audio signals, the use of the spread spectrum using pseudo random sequence [1], phase coding using all-pass filtering [2] and echo hiding [3] have been investigated.

In the data hiding technique using spread spectrum, digital information is encoded by superimposing random sequences. If the power level of the random sequences is below the perception level throughout the signal band, the distortion is expected to be imperceptible. From the viewpoint of decoding, on the other hand, there is a trade-off between the length of random sequences and the relative power level to the source signal.

In general, long random sequences, which limit the data rate of hiding, are required to keep the distortions imperceptible. In order to improve the data rate of hiding, it seems to be an effective approach to shape the spectrum of random sequence up to the imperceptive level of the source signal. The method using the masking characteristics of the auditory system [4][5] is one of the effective approaches on this direction.

In this paper, a novel method to embed secret speech into open speech is proposed. The secret speech is coded into binary parameter bits with Mix-Excitation Linear Prediction (MELP) algorithm, and the bits are used to form hiding information sequence.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 414–420, 2004.
© Springer-Verlag Berlin Heidelberg 2004

The open speech is automatically divided into voiced frames and unvoiced frame using auditory wavelet transform. One voice frame, the auditory wavelet transform was used to detect pitch, and the pitch is utilized to the current embedding position in open speech. The information hiding procedure is completed by modifying relevant wavelet coefficients. At the receiver, based on the same pitch detection method, the embedding position is found and the hiding bit is recovered. The secret speech can be received after MELP decoding. The experiments show that the method is strongly robust to many attacks such as compression, filter and so on.

2 Creating Secrecy Information

Secret speech was coded by MELP before embedded. MELP is robust in difficult background noise environments such as those frequently encountered in commercial and military communication systems. It is very efficient in its computational requirements. This translates into relatively low power consumption, an important consideration for portable systems.

The MELP is based on the traditional LPC parametric model, but also includes four additional features. These are mixed-excitation, a periodic pulses, pulse dispersion, and adaptive spectral enhancement. The mixed-excitation is implemented using a multi-band mixing model. This model can simulate frequency dependent voicing strength using a novel adaptive filtering structure based on a fixed filter bank. The primary effect of this multi-band mixed-excitation is to reduce the buzz usually associated with LPC vocoders, especially in broadband acoustic noise.

3 Pitch Detection Based on Auditory Wavelet

A pitch detector is basically an algorithm that determines the fundamental pitch period of an input speech signal. Pitch detection algorithms can be divided into two groups: time-domain pitch detectors and frequency-domain pitch detectors. Pitch detection of musical signals is not a trivial task due to some difficulties such as the attack transients, low frequencies, and high frequencies. The autocorrelation function is a time-domain pitch detector. It is a measure of similarity between a signal and translated (shifted) version of itself. The basic idea of this function is that periodicity of the input signal implies periodicity of the autocorrelation function and vice versa. For non-stationary signals, short-time autocorrelation function for signal $x(n)$ is defined as [7]:

$$T_l = \frac{1}{N}\sum_{n=0}^{N-m-1}[f(n+l)w(n+l)][f(n+m+l)w(n+m+l)] \quad 0 \le m \le N-1 \tag{1}$$

Where w (n) is an appropriate window function, N is the frame size, l is the index of the starting frame, m is the autocorrelation parameter or time lag 0 and M is the total number of points to be computed in the autocorrelation function. The autocorrelation function has its highest peak at m=0 which equals to the average power of the

input signal. For each l, one searches for the local maxima in a meaningful range of m. The distance between two consecutive maxima is the pitch period of the input signal $x(n)$. Different window functions such as rectangular, Hanning, Hamming, and Blackman windows have been used in the analysis. The choice of an analysis window and the frame size are among the main disadvantages of the autocorrelation function.

3.1 Auditory Wavelet

Because pitch detection (and hence f0 estimation) is, by its nature, a perceptual problem, any algorithm designed specifically for pitch should be able to be improved by adding some characteristics of the human auditory system. A simple improvement that can be added to any frequency-domain method is to use a constant-Q spectral transform instead of a basic Fourier spectrum. A constant-Q transform is more computationally demanding, but is more faithful to the human auditory perceptual system. Two factors must be considered when deciding whether or not to use human auditory modeling. First, the application for which the detector be used. If the goal is simply to detect the fundamental frequency of the signal without consideration of the pitch, human perceptual factors are probably not very important. However, if the goal is to detect the pitch for a transcription application, human factors are more relevant. The second factor is computational complexity. Human auditory modeling often results in a significant increase in the computation time required for the application. If computation time is a domain constraint, it may be necessary to forego auditory modeling in favor of a method, which is faster but less physiologically accurate. If properties of the human auditory system are to be used in any application, including f0 estimation, we must first understand the human perceptual system much better than we currently do. Presently, the most we can do is make the computer system provide the same type of results that the human system does, and hope that these improvements will make the system more accurate and robust [6-7].

Wavelet transform is based on the idea of filtering a signal $x(t)$ with dilated and translated versions of a prototype function $\psi(t)$. This function is called the mother wavelet and it has to satisfy certain requirements [8]. The *Continuous Wavelet Transform* (CWT) for $x(t)$ is defined as [9]:

$$CWT(f,a,b) = \int_{-\infty}^{\infty} x(t)\psi_{ab}(t)dt \tag{2}$$

Where $\psi_{ab}(t) = \psi(\frac{t-b}{a})$, $a \in R - \{0\}$ is the scale parameter and $b \in R$ is the translation parameter. In addition to its simple interpretation, the CWT satisfies some other useful properties such as linearity and conservation of energy. For practical implementations, CWT is computationally very complex. *Dyadic Wavelet Transform* (DWT), is the special case of CWT when the scale parameter is discretized along the dyadic grid (2^j), $j = 1,2\cdots$ and $b \in Z$

$$DWT(x,j) = W_j x = x(t) * \psi_{2^j}(t) \tag{3}$$

Where * denotes convolution and $\psi_{2^j}(t) = \psi(\frac{t}{2^j})$ for an appropriately chosen wavelet, the wavelet transform modulus maxima denote the points of sharp variations of the signal. This property of DWT has been proven very useful for detecting pitch periods of speech signals [10].

The auditory wavelet means that the mother wavelet is described by Gammachirp filter as flow:

$$gc(t) = at^{n-1} \exp(-2\pi bERB(f_r)t) \exp(j2\pi f_r t + jc \ln t + j\varphi) \tag{4}$$

Where time $t > 0$

a : amplitude

n and b : parameters defining the envelop of the gamma distribution

f_r :the asymptotic frequency

c :a parameter for the frequency modulation or the chirp rate

φ :the initial phase

$\ln t$:a natural logarithm of time

$ERB(f_r)$: equivalent rectangular bandwidth of an auditory filter at f_r

3.2 Voiced/Unvoiced and Pitch Detection Algorithm [11]

Step 1. Auditory wavelet transform based speech analyzer 17 scales with ERB distribution.

Step2. Maxima detector (with prefixd threshold per scale).

Step3. Peaks combination onto the first pitch period sequence $C_i(k)$, i is scale and k is the sequence number of peaks. $T_p = \arg C_i(k)$, So T_p is the auditory wavelet coefficient corresponding to $C_i(k)$.

Step4.window centered at each peak (with its width adapted to the pitch period value).

Step5. Voiced/unvoiced decision algorithm.

4 Embedded Secrecy Information

Step1. Using MELP, the embedded speech s was coded into secret information I which length is P bits.

$$I = \{x(i), 0 \le i \prec P\}, x(i) \in \{0,1\} \tag{5}$$

Step2. Base on the algorithm 2.2, the open speech S was segmented into voiced and unvoiced speech.

Step3. For the voiced speech, find the pitch period sequence $C_i(k)$ and T_p . The position is the embedded position.

$$cf_i^{'}(i)=\begin{cases} cf_i(j)(1+\beta\cdot x(l)), & j=T_p-2,T_p-1,T_p,T_p+1,T_p+2 \\ cf_i(j), & others \end{cases} \tag{6}$$

Where β is the embedded depth. $cf_i(j)$ is the wavelet coefficient. i is scale.

We use 5 points for embedding information to increase robust.

Step4. Using inverse wavelet transform, reconstruct the open speech S' based on $cf'_i(j)$.

5 Detecting and Extracting Secrecy Information

In this paper, it needs original speech to extract embedded information. Because the embedded position is unfixed, extracting information must search the pitch and the embedded position T_p. We can use the algorithm 3.2 to search the embedded position T_p just as the processing of embedding.

$$t_k=\frac{1}{5}[t_k(T_p-2)+t_k(T_p-1)+t_k(T_p)+t_k(T_p+1)+t_k(T_p+2)] \tag{7}$$

$$t_k(j)=\frac{1}{\beta\times cf_i(j)}(cf_i^{'}(j)-cf_i(j)) \tag{8}$$

Extracting $x(k)$:

$$x(k)=\begin{cases} 1, t_k>Th \\ 0, \quad others \end{cases} \tag{9}$$

Where Th is a threshold, which the users set.

After MELP decoded, the secret speech I' can be obtained.

6 Analyzing Performance

In experiment, we use the short-wave communication speech as the open speech, which sample is 8KHz. In this paper we use segment Itakura distance to measure the timbre of speech.

A_k is the LPC-10 feature vector of the k^{th} frame of signal I : $A_k=[a_0,a_1,\cdots a_{10}]$;

B_k is the LPC-10 feature vector of the k^{th} frame of signal I' : $B_k=[b_0,b_1,\cdots b_{10}]$;

R_k is self-correlative matrix of $I(k)$:

$$R_k = \begin{bmatrix} r_k(0) & r_k(1) & r_k(2) & \cdots & r_k(10) \\ r_k(1) & r_k(0) & r_k(1) & \cdots & r_k(9) \\ r_k(2) & r_k(1) & r_k(0) & \cdots & r_k(8) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ r_k(10) & r_k(9) & r_k(8) & \cdots & r_k(0) \end{bmatrix} \tag{10}$$

Where $r_k(m) = \sum_{n=0}^{L-m-1} x_k(n) x_k(n+m)$, L is the length of frame.

The Itakura distance between $I(k)$ and $I'(k)$ is as flow:

$$d_k(A_k, B_k) = \ln\left\{\frac{A_k R_k B_k^T}{A_k R_k A_k^T}\right\} \tag{11}$$

So, the average Itakura distance is $DI = \frac{1}{N}\sum_{k=1}^{N} d_k(A_k, B_k)$. Where N is the frame number of secret speech I.

To illuminate the robust of the algorithm, we take some measure to attack embedded speech.

(1) Compressed attack: compress the speech S' witch include secret speech using G.723 24/40kb/s ADPCM.
(2) Low pass filter: process the signal S' using a low pass filter which close frequency is 300Hz and 3000Hz respectively.
(3) color noise attack: use the cooledit software to create pink noise to add the signal S'.
(4) denoising attack: denoise the signal S' using DB5 wavelet.
(5) prune attack: prune the the signal S' into one fifth S'.

The Itakura distance after attacked show in table 1.

Table 1. Attack and Itakura distance

Attack	Itakura distance
Compress	0.0675
Low pass	0.0043
Color noise	0.0767
Denoise	0.1023
Prune	0.0186

7 Conclusions

This paper proposes a new method to embed secret speech into open speech. And test the robust of the algorithm. A lot of experiment show that the method perform well very for some attack such as compress, low pass, color noise, denoise and so on. This method fits not only to copyright protection but also secret communicates.

References

1. Ingemar J. Cox, Joe Kilian, Tom Leighton, and Talal Shamoon.: Secure spread spectrum watermarking for images, audio and video. ICIP'96, 1996, 234–246.
2. Laurence Boney, Ahmed H. Tewfik, and Khaled N. Hamdy.: Digital Watermarks for Audio Signals. IEEE Intl. Conf. on Multimedia Computing and Systems, Hiroshima, 1996,473–480
3. Laurence Boney, Ahamed H. Tewfik, and Khaled N Hamdy.: Digital Watermarks for Audio Signals. European Signal Proc. Conf., Trieste, Italy, September 1996, 1203-1210.
4. Yasemin Yardimci, et al.: Data hiding in speech using phase coding. ECSA. Eurospeech97, 1997, 1679–1683.
5. D. Gruhl, A. Lu, and W. Wender.: Echo hiding. Lecture Notes in Computer Science, Vol. 1174, Information Hiding, 1998. 295–315
6. John M. Eargle.: Music, Sound and Technology. Van Nostrand Reinhold, Toronto, 1995,464-470.
7. Albert Bregman.: Auditory Scene Analysis. MIT Press, Cambridge, 1990.
8. Kronland Martinet, R., Morlet, J. and Grossman, A.: Analysis of sound patterns through wavelet transforms. International Journal of Pattern Recognition and Artificial Intelligence, 1987,1(2): 273-302
9. Sadowsky, J.: The continuous wavelet transform: a tool for signal investigation and understanding. Johns Hopkins APL Technical Digest, 1994,15(4): 306-318.
10. Kadambe, S., and Boudreaux-Bartels, G.: Application of the wavelet transform for pitch detection of speech signals. IEEE Trans. on Information Theory, 38(2): 917-924, (1992)
11. Leonard Janer.: Modulated Gaussian wavelet transform based speech analysis pitch detection algorithm. In Processing of EURSIPCO, 1995, (1): 401-404

Automatic Selecting Coefficient for Semi-blind Watermarking

Sung-kwan Je[1], Jae-Hyun Cho[2], and Eui-young Cha[1]

[1]Dept. of Computer Science, Pusan National University,
[2]Dept. of Computer Information, Catholic University of Pusan
jimmy@harmony.cs.pusan.ac.kr

Abstract. In this paper, we present a watermarking scheme based on the DWT (Discrete Wavelet Transform) and the ART2 to ensure the copyright protection of the digital images. The problem to embed watermark is not clear to select important coefficient. We used the ART2 to solve it. We didn't apply the whole wavelet coefficients, but applied to only the wavelet coefficients in the selected cluster. Disadvantage of ART2 is different train data according to sequence of input data, but it becomes advantage of watermarking. Using the ART2 that even the watermark casting process and watermark verification process are in public, nobody knows about the location of embedding watermark except of authorized user. As the result, the watermark is good at the strength test-filtering, geometric transform and etc.

1 Introduction

Several studies protecting the copyright of digital multimedia contents have been proposed [1-5]. In this situation, the studies of copyrighting multimedia contents have been proposed in many ways. The most important character of digital watermarking is imperceptible. If the watermark is embedded, viewers should not see nor notice the mark. If the information of image is being distributed illegally, we can trace the flow of the data using the embedding the watermark in the information of the image. We can find the man who has distributed it.

Recently, digital watermarking has been classified by two ways; Spatial Domain and Frequency Domain. In the spatial domain, the watermark is embedded directly in the spatial domain. In this process, various researches have been developed as PN-Sequence (Pseudo random Noise Sequence) [6] and statistical method, etc. The process to embed the watermark in spatial domain is simple and fast but it has disadvantaged that it's weak from the external attack, noise and JPEG compression.
Because of the result, the study of the watermarking is mainly researched in the frequency domain in recent. The process of watermarking in the frequency domain is that the watermark is embedded in the repetitive and characteristic coefficient among the generated coefficients which are transformed from FFT, DCT, Wavelet, and etc. Cox [4-5] proposed the process of watermarking using the DCT. In the process, signal spread of frequency domain is widely distributed to transmit effectively the

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 421–430, 2004.
© Springer-Verlag Berlin Heidelberg 2004

watermark signal without noise; filtering, compression and transform using spread spectrum communication. The energy of the specific signal spread is too small to be noticed, but the signal is extracted by PSNR (Peak Signal to Noise Ratio) using the location and variation of the original image signal. The problem of the process is not clear to select important coefficient, and partly characters can not be effective because of transforming the whole image. It is hard to select important coefficients for embedding watermark using the DCT. Further, the watermark embedded by the robustness of JPEG compression is easily loss because compression used the 8•8 block DCT. Besides, at this scheme, if it does block transformation, it could be happened the Block phenomenon. So it brings the loss of image.
This paper is organized as follows; we propose the watermarking scheme in section 3 and the experimental result and conclusion at the final section.

2 ART2 (Adaptive Resonance Theory)

In usual, when the watermark is embedded in a certain image, it is important to decide the location where it is embedded. It is difficult to select the location that has robustness from the attack. The extract of this region is generally expressed to ROI (Region of Interest).
There are various algorithms in the ANN algorithm. In this paper, we apply the ART2 algorithm among ANNs for ROI. Each algorithm has each application field. In general, when the purpose is to classify the cluster of the data in the characteristic similarity, the applications; ART, SOM and Fuzzy-ART, should be used in large. We use the ART2 algorithm that has less limitation of the size of the input data. The ART2 is proposed the ANN model by Grossberg and Capenter, and have the characteristics outlined below [11].

- Adaptability; when data added, not necessary to training trained data again.
- Stability; It designed not to lose the plasticity to training new pattern.
- Real-time processing; The processing is fast and stable about the unlimited input data

The ART2 can be applied in binary input pattern and analog input pattern. Furthermore, the connection weight of variation of the ART2 gets the average of the whole input pattern, so it responses equally to generate the cluster. We use generated the character of similar pattern cluster of the ART2 to decide the location to embed the watermark. The ART2 can select the location of cluster where users want to embed, so it has a strong point that the embedding location can be controlled by the character of the image. Disadvantage of ART2 is different train data according to sequence of input data, but it becomes advantage of watermarking. Nobody knows about the location of embedding watermark except of authorized user. If they processed same ART2, the trained data would different from the result of ours.

3 Proposed Algorithm

3.1 Embedding Watermark

When the image data transforms to frequency domain if it is communication channel, the watermark is that signal is transmitted by the communication channel. The signal should not be affected by noise, filtering, compression and transmission during the transmission. We use the signal of the watermark as Gaussian normal distribution: the average is 1, and variance is 1. Gaussian random vector is invisible when it is embedded, and it's also stronger than the binary watermark [4].
We decompose the original image with the DWT to 3-levels of MRA. In most natural images, the energy is concentrated on the lower frequency domain that relates with human vision. If the image data damaged in the lower frequency domain, people could have noticed about it. Accordingly, for protecting the quality of the image and making the watermarked image, we embed the watermark at the highest frequency domain where a little information of images is in. Using the DWT, embedding the watermark calculated the threshold to embed the watermark with equation (2) to calculate the threshold which decides to embed each sub-band.

$$T = 2^{\lfloor \log_2 MAX(Wavelet\ Coefficient) \rfloor} \tag{2}$$

It is the one of the image compression algorithm that Shapiro [12] proposed to calculate the threshold which used zero padding in the EZW (Embedded zerotree wavelet algorithm). EZW method is very efficient to encode important wavelet coefficient and express the energy concentration phenomenon by using the wavelet transform. It is to embed the watermark that used to calculate the first threshold. But, It is to embed the watermark which a several bigger wavelet coefficient using the maximum of sub-band. So, it couldn't express the characters of the image nor adjust the length of the watermark. If we adjust the vigilance, we will adjust the amount of the watermark in the same image. So we can say that it is strong for the image procession like compression or cropping.

In this paper, we classify wavelet coefficients of the highest sub-band (LH, HL and HH) using the ART2. The ART2 creates the clusters dynamically, so it is not influenced by the size of the image. We also can adapt various images to it. In the case of the SOM which is one of other competitive learning proposed by Kohonen, it sets the number of clusters in advance, so that it is influenced by the size of the image. For example, in the case of a test image, the number of wavelet coefficients in a cluster changes following to the change of image size. And that of Lena image which have big wavelet coefficients, the number of wavelet coefficients in a cluster is more than that of others. As so, before considering of the characteristics of image, we can't adjust the number of the watermark. We can say the ART2 is adaptable to the image because it is not influenced by the size of the image. And we could adjust the amount of watermark in the same image and could do that following the specialties of each image.

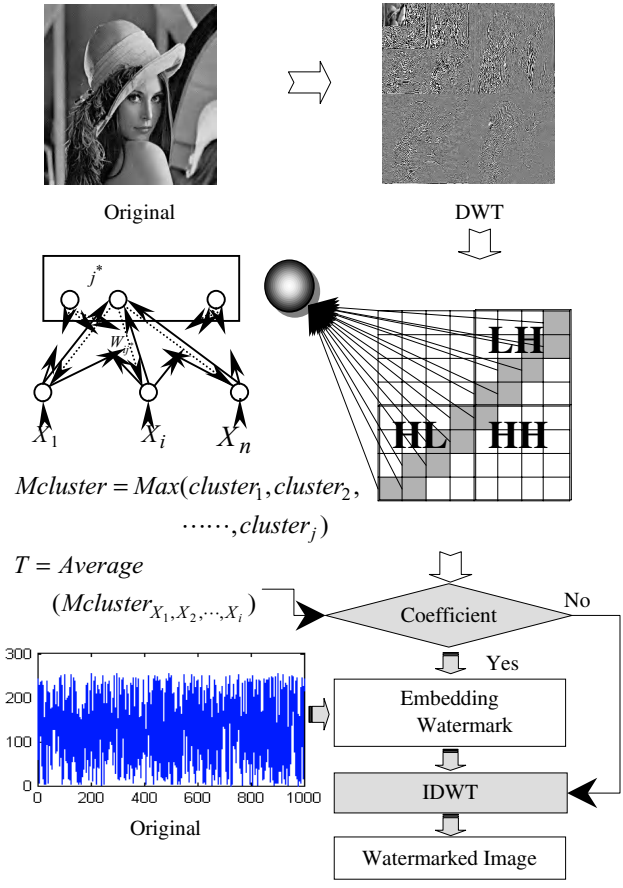


Fig. 1. Embedding watermark (X_i is a wavelet coefficient in the highest sub-band, W_{ij} is a connection weight, and *j* is a cluster)

$$X_i^{'} = X_i + \alpha W_i \tag{3}$$

$$X_i^{'} = X_i\left(1 + \alpha W_i\right) \tag{4}$$

$X_i^{'}$ is a watermarked coefficient, X_i is a wavelet coefficient, and W_i is a watermark. When we embed watermark to the image to obtain the watermarked image using equation (3) and (4). Equation (3) is just adding the watermark, so it is not proper when the variation of the value has extreme differences.

In equation (4), the variation of a scaling parameter affects largely the embedding watermark. As a result, we used equation (4), and value is 0.6, 0.8 and 1 according to variance of coefficients.

As we tested, we set the vigilance to 0.05 in the ART2 algorithm, and we select the biggest cluster among the classified clusters. We set the average to threshold and we

embed the watermark to the coefficient what is bigger than that of the average in those of the biggest cluster in figure (1).

3.2 Extracting Watermark

We decompose the watermarked image like the processing of embedding watermark to 3-level using the DWT. We calculated the information of the biggest cluster which classified by using the ART2 in the embedding processing.

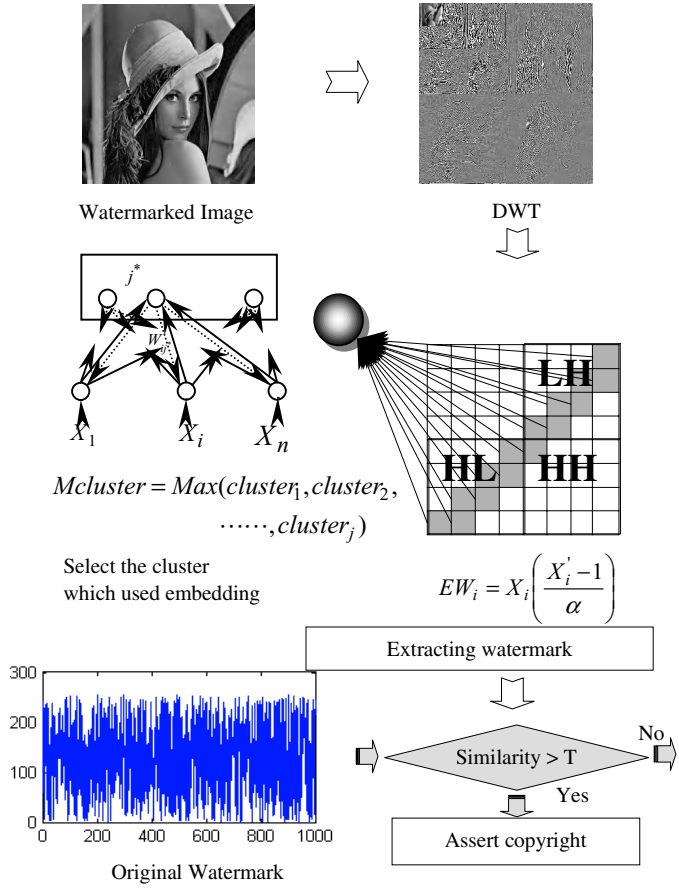


Fig. 2. Extracting watermark

It use for extracting the watermark without original image. The information has a location and average of embedding the wavelet coefficient. Unauthorized users know about location of embedding the watermark, and the watermark will remove easily. So, in this paper, nobody knows about the location of embedding watermark using the ART2. Even the watermark is embedded using the ART2 and the watermark

verification process is in public, unauthorized users don't know about the information of the trained data. If they processed same ART2, the trained data would different from the result of ours. As a result, the algorithm is safer than others.

For evaluating similarity, there are some schemes. One is the way of calculating Vector projection, other is the way of calculating correlation, other is the way of calculating Bit error, and etc [4].

$$Correlation\left(X, X^*\right) = \frac{\sum XX^*}{\sqrt{\sum X^2 \sum X^{*2}}} \tag{5}$$

X is a original watermark, X^* is a extracted watermark. In this paper, we use the equation (5) for evaluating similarity between two vectors. If the similarity between the original watermark and the extracted watermark is higher than a threshold, we could assert the copyright.

4 Experimental Results

In this paper, we experiment with Pentium 700 MHz, Window XP and Matlab 5.2. The size of the image is 256•256, and we test various image such as Lena image, Barbara image, Bridge image and Girl image, and etc. We use the watermark as the Gaussian normal distribution: average is 1, and variance is 1.

We experiment fidelity and robustness for the standard of the performance value. For the test of robustness, we did various filtering (Lowpass Filter, Highpass Filter, Wiener Filter), adding noise, geometric transform (enlarge, reduction, cropping) and the attack of the compression of JPEG, and then we confirm robustness. In addition, for the higher confidence in the proposed algorithm, we test the image which is not embedded watermark by false positive error. We also are compared with other algorithms (Kundur, Wang, Xia, Cox and Kutter).

4.1 Similarity

We get PSNR to be decided objectively between the original image and the image embedded watermark, and we calculate fidelity through the equation (7) from the extracted watermark and the original watermark in the table 1, the value of objective PSNR is maintained over 45dB.

4.2 Robustness

We tested the watermarked image in Lowpass filer, Highpass filter, and Wiener filter. We used Highpass filter that the mask of the 3 x 3 size, [0 -1 0; -1 8 -1; 0 -1 0]/4, and

(a) Original image (b) Watermarked image

Fig. 3. Similarity Test

Table 1. Similarity between orignal image and watermarked image

Image	PSNR	Image	PSNR
Lena	47.92	Camera man	47.92
Barbara	47.35	Crowd	46.57
Bridge	47.54	Oleh	46.85
Girl	47.75	Pepper	47.35

Lowpass filter that the 3 x 3 size of the Gaussian filter, average is 0, and standard deviation is 0.5, and Wiener filter is the 3 x 3 size of the Wiener filter. Wiener filter is less similarity than other filters but it is not influenced to extract the watermark. We tested the watermarked imaged about geometric transform (rescaled a twice enlarged the watermarked image, rescaled a twice reduced the watermarked image, and the 156•156 size of the center cropping). The result is powerful efficiency in the geometric transform and adding noise of Salt & Pepper and Gaussian.

Table 2. Correlation between the original watermark and the extracted watermark

Image	Lowpass filter	Highpass Filter	Wiener filter	Rescaled enlarge	Center cropping	S&P noise
Lena	0.99	0.98	0.70	0.98	0.88	0.86
Barbara	0.99	0.97	0.75	0.97	0.84	0.86
Bridge	0.99	0.99	0.76	0.99	0.82	0.89
Girl	0.98	0.99	0.75	0.98	0.92	0.83

4.3 Compare with Others

We also compare with other watermarking using PN-Sequence. In the frequency domain, the original image needed Wang [8] and Xia [9] using the wavelet transform, Kundur [7] without the original image, the original image needed Cox [4] using the DCT, Kutter's [6] algorithm embedded watermark using the spatial domain.

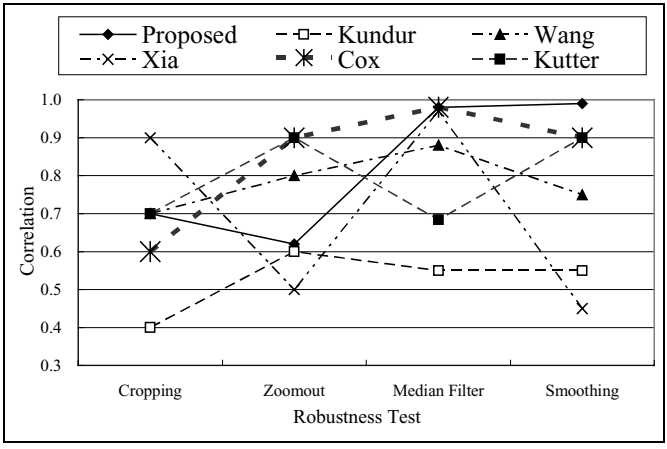


Fig. 4. Compare robustness with others (Correlation between the original watermark and the extracted watermark)

In the test of robustness such as cropping and filtering, the result is powerful efficiency. In the geometrical transformation such as reduction, the process is less similarity than other algorithm; Wang, Cox and Kutter.

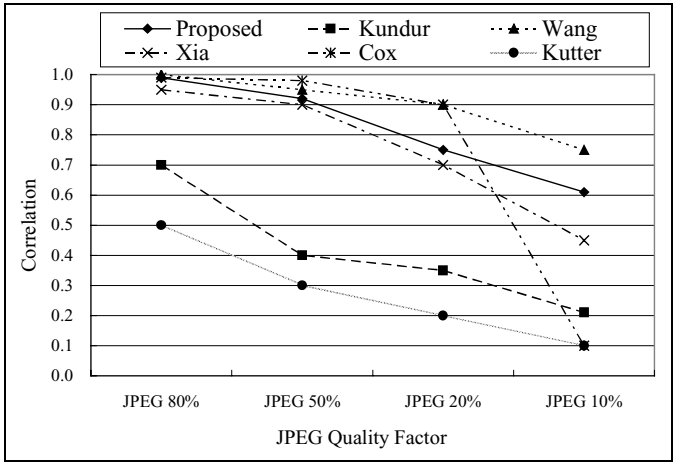


Fig. 5. Compare JPEG compressing with others

But, it doesn't matter to decide the existence of watermark because the similarity is over than detection values. However, it is not influence to extract watermark. Compare with Kundur's algorithm which doesn't need the original image, the process get better result in figure (4).

Even though the efficiency of the process is weaker than other algorithms which needed the original image but it is not influence to extract the watermark. In this paper, without the original image, so the process is better than other blind

watermarking in comparison. Furthermore, compare with Cox in the same the DCT situation, it can't be extracted under JPEG 10%, the watermark is extracted in the process even the value is low.
The image that is not embedded the watermark is experimented in false positive error. The watermarking algorithm can't be reliable, if the watermark is extracted in false positive error. In this paper, the watermark isn't extracted from the image that not embedded the watermark.

5 Conclusion

In this paper, we propose the watermarking considering of human vision character and embed the watermark in the highest sub-band that has fewer amounts of image data in visual. The process uses the wavelet transform by using the ART2. The process considers the character of the image that is adaptive watermarking. Using the clustering data that is used in embedding, the watermark is extracted without the original image. We not applied the whole wavelet coefficients, but applied to only the wavelet coefficients in the selected cluster to reduce the time cost. The algorithm is much stronger than the others because unauthorized users can't know the result of training by the ART2.
In the result, the value of objective PSNR is maintained over 45dB, and there is not to significant visual difference in subjective observation. And the proposal algorithm is much efficient than other algorithms.

References

[1] M. D. Swanson, M. Kobayashi, and A. TewFik, "Multimedia Data-Embedding and Watermarking Technologies," *In Proceeding of IEEE*, Vol. 86, No. 6, June 1998.
[2] I. Pitas and T. Kaskalis, "Applying Signatures on Digital Images," *In Proceeding of IEEE Nonlear Signal Processing Workshop*, Thessaloniki, Greece, 1995.
[3] C. F. Osborne, R. G. Schyndel and A. Z. Tirkel, “A Digital Watermarking," *International Conference on Image Processing*, November 1994.
[4] I. J. Cox, J. Kilian, T. Leighton and T. Shamoon, "Secure Spread Spectrum Watermarking for Multimedia," *IEEE Transaction on Image Processing*, Vol. 6, No.12, pp.1673-1687, 1997.
[5] I.J. Cox, J. Kilian, T. Leighton and T. Shamoon, "Secure Spread Spectrum Watermarking for Images, Audio and Video," *International Conference on Image Processing*, Vol. 3, pp.243-246, 1996.
[6] M. Kutter, F. Jordan, and F. Bossen. "Digital Signature of Color images using Amplitude Modulation," *In Ishwar K. Sethi, editor, Proceedings of the SPIE Conference on Storage and Retrieval for Image and Video Database*s, Vol. 2952, pp. 518 - 526, San Jose, USA, 1997.
[7] D. Kundur and D. Hatzinakos, "Digital watermarking using Multiresolution Wavelet Decomposition," *In Proceedings of IEEE ICASSP '98*, Vol. 5, pp. 2969 - 2972, Seattle, WA, USA, May 1998.
[8] H. J. Wang, P. C. Su and C. J. Kuo, "Wavelet-based digital image watermarking," *Optics Express*, 3 pp. 497, December 1998.

[9] X. G. Xia, C. G. Boncelet and G. R. Arce, “Wavelet Transform based Watermark for Digital Images," *Optics Express 3*, pp. 497, December 1998.
[10] S. Mallat, "Multi-Frequency Channel Decomposition of Images Wavelets Models," *IEEE Trans. on Information Theory*, Vol. 11, no. 7, July 1992.
[11] S. Haykin, Neural Networks: A Comprehensive Foundation, *MacMillan*, 1994.
[12] J. M. Shapiro. "Embedded Image coding using zerotrees of wavelet coefficients," *IEEE Trans. on Signal Procsseing*, Vol. 41, No. 12, pp. 3445-3462, December 1993.

Network Probabilistic Connectivity: Optimal Structures*

Olga K. Rodionova[1], Alexey S. Rodionov[1], and Hyunseung Choo[2]

[1] Institute of Computational Mathematics and Mathematical Geophysics
Siberian Division of the Russian Academy of Science
Novosibirsk, RUSSIA +383-2-396211
rok,alrod@rav.sscc.ru
[2] School of Information and Communication Engineering
Sungkyunkwan University
440-746, Suwon, KOREA +82-31-290-7145
choo@ece.skku.ac.kr

Abstract. The problems of optimizing the network structure by the reliability criteria is discussed. The networks with absolutely reliable nodes and unreliable edges are considered, and a special attention is given to the structures based on rings. The tasks of global optimization, optimal interconnection, and optimal addition of new edges to existent graphs are considered and reliability polynomials are used for optimization. Some derivations are made with the use of original methods based on consideration of long chains.

1 Introduction

Optimizing the network structure by reliability criteria is well-known task but unsolved in general. Networks with equal reliability of edges and absolutely reliable are studied in the current paper. We use the probability of all-nodes connectivity as a reliability index that is rather common [1,2,3,4,5,6]. Further we refer to it as simply *reliability*. In [1,7] we can find the branch-and-bound algorithms for constructing a most reliable structure with or without the cost constraint with limited or given number of elements. In [2] the algorithms for optimal addition of elements to a given network are considered. In [8] some theorems are proven about optimal structures of graphs with number of nodes n and number of edges $n-1, n$ and $n+1$. The work [9] contains some preliminaries and atlas of uniform graphs of small dimension with their reliability polynomials. The paper [10] contains the rules for optimal interconnection of 2 cycles and optimal addition of two edges to a cycle. We consider the optimal inter-connection or the optimal addition of new edges to existent networks based on *circular structures* or *rings*. As it is known, circular network structures are made up with cycles connected by a relatively small number of edges or chains, or from one cycle with a small

* This paper was partially supported by BK21 program, University ITRC and RFBR. Dr. Choo is the corresponding author.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 431–440, 2004.
© Springer-Verlag Berlin Heidelberg 2004

number of additional edges, or a mixture of these two types. Such graphs, for example, can be used as a model for optical networks based on SONET rings.

The rest of the paper is organized as follows: in section 2 we present the preliminary results needed for further considerations. Section 3 contains the discussion of the optimal development of networks while section 4 is devoted to the optimal interconnection of graphs.

2 Preliminaries and Tools

First let us make the following denotations:

$G(n,m)$, $C(n)$, $T(n)$ – a non-oriented graph with n nodes and m edges, a cycle or tree with n nodes, respectively;
$R(G)$ – probability of connectivity or reliability of a graph G;
$R(p)$, $R(G,p)$ – a reliability polynomial, the name of a graph is included as an argument if needed, p is a reliability of an edge.

The number of spanning trees is usually used as a reliability criteria [4], yet this index can lead to wrong conclusions. In [9] the example of two graphs G(6,11) is presented (see Fig. 1). There are 224 spanning trees in the first and 225 in the second graph in this figure. Yet the first graph has a greater probability of connectivity for the edge reliability less than $1 - \sqrt{2}/2 \leq p \leq 1$ as is shown later. We usually use a reliability polynomial [9,11] as a tool for comparison of

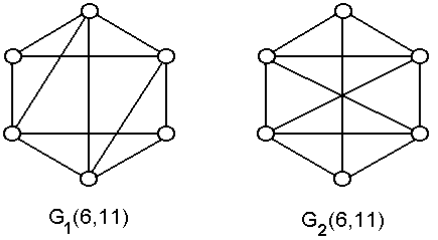


Fig. 1. Example of graphs optimal at different value of an edge reliability

different structures. In the case of equal edge reliability p a reliability function or a reliability polynomial shows the reliability of a network. This polynomial is better to present in the following way:

$$R(p) = p^m + \sum_{i=1}^{m-n+1} a_i (1-p)^i p^{m-i}, \tag{1}$$

that corresponds to the expansion by connected sugraphs with given number of edges (i edges fail, $m-i$ exist). As "sugraph" we name a subgraph that includes the complete set of nodes. Thus, for the graphs in Fig. 1 we have the coefficients

(1, 11, 55, 163, 310, 370, 224) and (1, 11, 55, 163, 309, 368, 225), respectively. The difference is

$$R(G_1,p)-R(G_2,p)=p^4(1-p)^{11}+2p^5(1-p)^{10}-p^6(1-p)^9=p^4(1-p)^9(1-2p^2), \tag{2}$$

that ascertains that these graphs are optimal on different intervals of edge reliability.

We use the well-known formula of branching [13]:

$$R(G)=p_{ij}R(G^*(e_{ij}))+(1-p_{ij})R(G\backslash\{e_{ij}\}), \tag{3}$$

where $G^*(e_{ij})$ and $G\backslash\{e_{ij}\}$ are graphs obtained by contracting an edge e_{ij} with a reliability p_{ij} or deleting it. We use two theorems that are proven in [14] and are re-formulated for the case of uniform edge reliability.

Theorem 1. *Let a graph G have a simple chain which consists of k edges $e_1, e_2, \ldots, e_k$, connecting nodes s and t. Then the reliability of G is equal to*

$$R(G)=p^k\cdot R(G^*(e_1,e_2,\ldots,e_k))+k(1-p)p^{k-1}\cdot R(G\backslash\{e_1,e_2,\ldots,e_k\}), \tag{4}$$

if an edge e_{st} directly connected nodes s and t does not exist and

$$R(G)=\left[p^{k+1}+(k+1)p^k(1-p)\right]\cdot R(G^*(e_1,e_2,\ldots,e_k)+ kp^{k-1}(1-p)^2\cdot R(G\backslash\{e_1,e_2,\ldots,e_k,e_{st}\}), \tag{5}$$

otherwise, where $G^*(e_1,e_2,\ldots,e_k)$ is a graph obtained from G by contracting by a chain $\{e_1,e_2,\ldots,e_k\}$, $G\backslash\{e_1,e_2,\ldots,e_k\}$ is a graph obtained from G by removal of this chain with nodes (except for terminal ones).

Theorem 2. *Let a graph $G_1(n,m)$ have a simple chain consisting of k edges $e_1, e_2, \ldots, e_k$, that connects nodes s and t. Then the reliability of $G_1(n,m)$ is equal to*

$$R(G_1(n,m))=\left[p^k+kp^{k-1}(1-p)\right]R(G_2(n-k+1,m-k+1)), \tag{6}$$

where a graph $G_2(n-k+1,m-k+1)$ is derived from $G_1(n,m)$ by substituting the chain by a single edge with the reliability

$$p=\left[p^k+kp^{k-1}(1-p)\right]/p^k. \tag{7}$$

Also we use the well-known expressions (see [8], for example) for reliability polynomials of tree and cycle and the rules for obtaining such polynomials in case of existence of dangling node, articulation point or a bridge in the network structure.

It is proven [12] that optimal graphs (except trees) must satisfy X- and U-uniformity. A graph $G(n,m)$ is X-uniform if the degree $\deg(v)$ of any node v in the graph satisfies

$$\delta(G)=\min_x \deg(x)\le\deg(v)\le\delta(G)+1, \tag{8}$$

and multi-graph $G(n, m)$ is U-uniform if the multiplicity $|u_i|$ of any multi-edge u_i in this multi-graph satisfies

$$\mu(G) = \min_i |u_i| \leq |u_i| \leq \mu(G) + 1. \tag{9}$$

3 Optimal Structures

3.1 Global Optimization

Case $m = n$**.** $C(n)$ is optimal for $G(n, n)$ by the reliability criteria from (8) and (9). It is proven in [8] without considering these criterias.

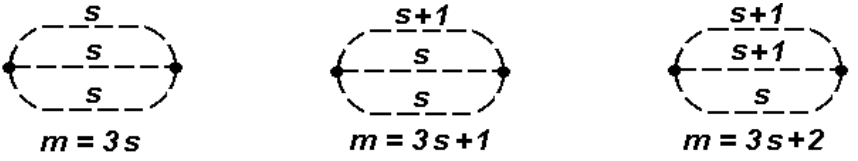


Fig. 2. Optimal graph structures when $m = n + 1$

Cases $m = n + 1$ **and** $m = n + 2$**.** In [9] the optimal graphs are presented for these cases. The optimal structures in these cases do not depend on the edge reliability. Case $m = n+1$ gives 3 variants of an optimal structure depending on the rest of a division $m/3$ (see Fig. 2) and the reliability polynomial for them is

$$R(G(n, n+1), p) = p^{n+1} + (n+1)(1-p)p^n + A(1-p)^2 p^{n-1}, \tag{10}$$

where A obtained by the formula

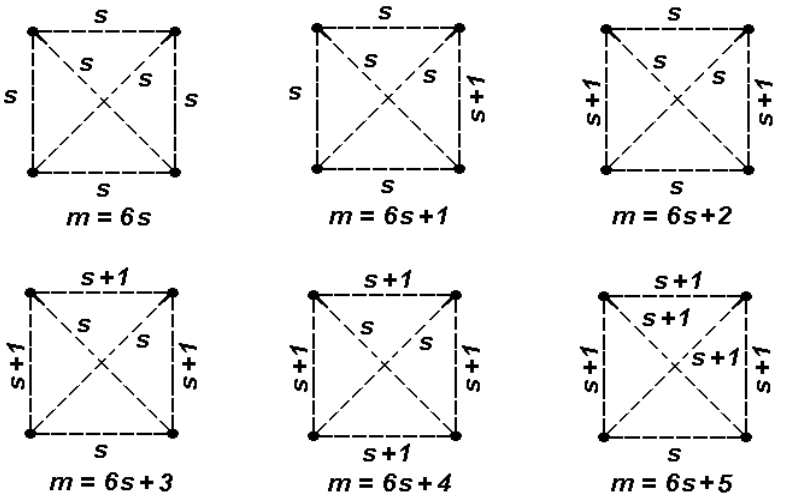
$$A = \left[\frac{n+1}{3}\right] \cdot \left[\frac{n+2}{3}\right] + \left[\frac{n+1}{3}\right] \cdot \left[\frac{n+3}{3}\right] + \left[\frac{n+2}{3}\right] \cdot \left[\frac{n+3}{3}\right]. \tag{11}$$

In [8] the case $m = n + 1$ is considered also but for one variant only and the common formula is not obtained. The case of $m = n + 2$ gives 6 variants of an optimal structure depending on the rest of the division $m/6$ (see Fig. 3).

General case. In general case there can be different optimal structures for different values of p for given n and m. Thus the only way is to renumber all X- and U-uniform $G(n, m)$ and to compare their reliability in the case of given p or compare their reliability polynomials for choosing optimal structures for different intervals of p variation.Usually it is impossible for a graph with dimension of practical interest.

3.2 Optimal Addition of Edges to a Cycle

Let us consider the task of the optimal addition of one edge (chord) to $C(n)$. In [8] the theorem is proven that the optimal is to place a chord so that its



.

Fig. 3. Optimal graph structures when $m = n + 2$

ends divide the cycle by half. The proof is based on the consideration of the reliability polynomial for the cycle $C(n)$ with a chord whose ends divide the cycle into chains with lengths k and $n - k$:

$$R(p) = pC(k)C(n-k) + (1-p)C(n) = p[p^k + kp^{k-1}(1-p)][(p^{n-k} + (n-k)p^{n-k-1}(1-p)] + (1-p)[p^n + np^{n-1}(1-p)] = [p^{n+1} + np^n(1-p) + k(n-k)p^{n-1}(1-p)^2] + (1-p)[p^n + np^{n-1}(1-p)]. \quad (12)$$

We show that if the reliability of the additional edge is $q \neq p$ then the conclusion will be the same. In fact, the result of branching in this case is

$$R(G(n, n+1)) = p^n + np^{n-1}(1-p) + q(n-k)kp^{n-2}(1-p)^2, \quad (13)$$

where the last term for any q and p is maximal when $k = [n/2]$.

There are two possible variants of addition two edges (chords) to $C(n)$ (see Fig. 4). Let the nodes incidental to additional edges divide the cycle on chains that consist from m_1, m_2, m_3, and $n - m_1 - m_2 - m_3$ edges, respectively.

Theorem 3. *For $C(n)$ with two additional edges (1) the crossed placement of them is better for any m_1, m_2, m_3 and $n - m_1 - m_2 - m_3$ and (2) the case is best when these lengths differs not more than by 1.*

The **proof** of the first statement is based on comparing the reliability polynomials of the considered graphs that are obtained by branching by chords. For the proof of the second statement we fix m_1 and m_2 or m_1 and m_3 in the reliability polynomial for the case of crossed chords and obtain that other two lengths must

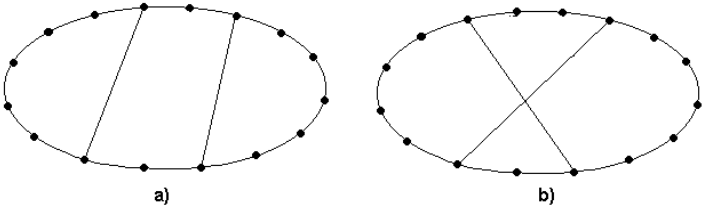


Fig. 4. Variants of addition two edges to a cycle

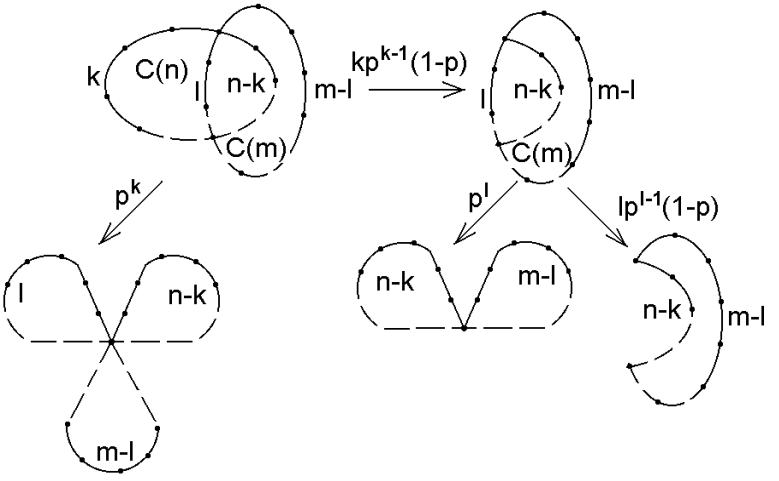


Fig. 5. Deriving reliability polynomial for the crossing of two cycles

be balanced. There are two variants for each fixation: odd or even number of edges in the both remaining chains. □

As in the case of one additional edge the addition of two edges with reliability different from p leads to the same rule when choosing the connected nodes.

4 Optimal Connection of Cycles

Optimal Crossing of Two Cycles Let us consider the crossing of two cycles $C(n)$ and $C(m)$. Let joint nodes divide the first cycle to chains with lengths k and $n-k$, and second – on chains with lengths l and $m-l$. The resulting graph consists of two nodes with degree 4 that are connected by these chains. Consequently applying the formulas from the theorem 1 to chains with lengths k and l after simple transformations we obtain the reliability polynomial (see Fig. 5):

$$R(p) = p^{n+m} + (n+m)p^{n+m-1} + [nm + l(m-l) + k(n-k)]p^{n+m-2}(1-p)^2 + [nl(m-l) + mk(n-k)]p^{n+m-3}(1-p)^3. \quad (14)$$

First two terms does not depend on the chain lengths while the third and forth are maximal in the case of balanced division of cycles. Note that if the reliability of edges differs in our cycles then the rule will be the same.

Optimal Cyclic Connection of Cycles.

Let us have k cycles with lengths $n_1, n_2, \ldots, n_k$, that are connected cyclically as is shown in the Fig. 6a. It is obvious that such a graph is X-uniform and U-uniform. The optimal mode for the placement of connecting edges is defined by the following theorem.

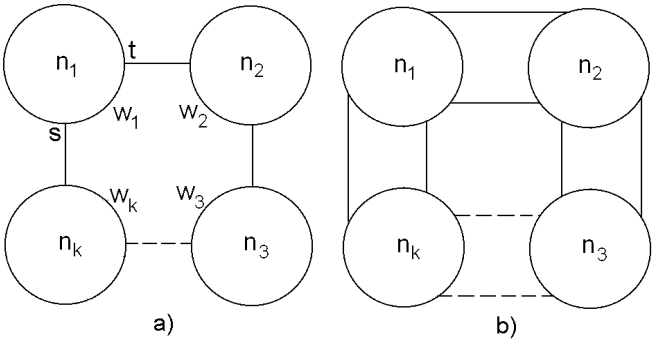


Fig. 6. Cycles of cycles

Theorem 4. *The probability of connectivity for a graph G which shows the cyclic connection of cycles has a maximal value when the nodes incidental to the connecting edges in any cycle divide the connected cycle into two chains with balances lengths.*

Proof. Let us choose any cycle. For better readability we omit index: let its length be equal to n and let nodes that are incidental to connecting edges (we denote them as s and t) divide the cycle onto chains with lengths w and $u = n-w$.

According the theorem 2 we substitute the chains with lengths w and u by edges with the reliabilities

$$p_1 = \frac{p}{w-(w-1)p}, \qquad p_2 = \frac{p}{n-w-(n-w+1)p}. \tag{15}$$

Then we change the obtained multi-edge by one with the reliability

$$p^{\circ} = p_1 + p_2 - p_1 \cdot p_2 = \frac{np-(n+1)p^2}{[w-(w-1)p][n-w-(n-w+1)p]}. \tag{16}$$

The obtained graph we denote as H and the newly obtained edge as e_{st}. The combined value of the factor on which we must multiply the reliability of H to obtain the reliability of G is, according the theorem 2:

$$r = p^{n-2}[w(n-w)-(n-2)p+(w-1)(n-w-1)p^2]. \tag{17}$$

Now we make one branching by this edge by the formula (3). Thus

$$R(G) = rR(H) = rp^{\circ}R(H^*(e_{st})) + r(1-p^{\circ})R(H\backslash\{e_{st}\}). \tag{18}$$

The graphs $H^*(e_{st})$ and $H\backslash\{e_{st}\}$ do not depend on dividing the cycle into chains. Let us compare the reliabilities obtained by the formula (18) for the cases of balanced and unbalanced division of the cycle into two chains. We refer to the correspondent reliabilities of edges that substitute chains, factors and reliability polynomial as p^*, r^*, and $R^*(p)$ in the balanced case, and p', r', and $R'(p)$ otherwise. For short we denote $R(H^*(e_{st}))$ as A and $(H\backslash\{e_{st}\})$ as B.

We have the difference

$$\begin{aligned} R^*(G) - R'(G) &= r^*p^*A + r^*(1-p^*)B - r'p'A - r'(1-p')B \\ &= (r^*p^* - r'p')(A-B) + (r^* - r')B. \end{aligned} \tag{19}$$

It is obvious that $(r \cdot p^{\circ})$ is always equal to $np^{n-1} - (n-3)p^n$ in our case, thus the first term in (19) is always zero. Thus the sign depends on the second term (in fact on $\delta = r^* - r'$).

1). Let $n = 2k$. In the presumptive optimal case $w = n - w = k$. The alternative is $w = k + d,\ 1 \le d < k$. From (17) we obtain:

$$\delta = p^{n-2}(k-kp+p)^2 - p^{n-2}[(k-kp+p)^2 + (d-dp)^2] = p^{n-2}(d-dp)^2 > 0. \tag{20}$$

2). Let $n = 2k+1$. In the presumptive optimal case $w = k,\ n - w = k+1$. The alternative is $w = k + d,\ 2 \le d < k$. In this case

$$\begin{aligned} \delta = p^{n-2}\left[(k-kp+p)^2 + (k-kp+p)(1-p)\right] - p^{n-2}\left[(k-kp+p) + \right. \\ \left.(d-dp)\right][(k-kp+p) - (d-dp) + 1 - p] = p^{n-2}d(1-p)^2(d-1) > 0. \end{aligned} \tag{21}$$

By applying this reasoning to all cycles we obtain proof of the theorem. □

As the **corollary** of the theorem proof we obtain its extension. As the consideration of one cycle does not affect to the properties of the rest of the graph, including probabilities of connecting edges, we have:

Theorem 5. *At the connecting of an arbitrary graph G by a pair of edges to a cycle $C(n)$ with uniformly reliable edges, when node(s) for connection in G are fixed, the optimal choice of two nodes in $C(n)$ is such that divide it into two chains of balanced lengths.*

Now let us have k cycles with lengths $n_1, n_2, \ldots, n_k$, that are connected cyclically by pairs of edges as is shown in Fig. 6b. This kind of graph is X-uniform and U-uniform if no one node is incidental to more than one connecting edge.

Theorem 6. *The optimal choice of nodes for connecting cyclically an arbitrary number of cycles with equally reliable edges and numbers of nodes $n_i > 3$ by pairs of edges with equal reliability p is such that all the cycles are divided by them onto the chains that differ not more then by one in each cycle.*

The proof of the theorem includes the consideration of 32 different variants and is out of the paper scope. The proving technique is similar to the one in the previous case. As in the previous case the theorem proving that deals with the only one cycle allows us to formulate a more general theorem.

Theorem 7. *The optimal rule for the choice of 4 nodes in the cycle $C(n)$ with equally reliable edges that are connecting with an arbitrary graph G is any four that divides $C(n)$ into 4 chains of lengths with difference within 1.*

It seems obvious that division of a cycle on chains with equal lengths is optimal for an arbitrary number of edges or chains that connect it with some graph G. Yet the question about the optimal choice of nodes in this graph is open.

5 Results of Experiments

For short we present only one but interesting example showing the significance of the structural optimization of networks. In Fig. 7 we present the growth of an average reliability of random graphs $G(16, m)$ with $m = 15, \ldots, 30$ and $p = 0.95$. For $m = 16, 17, 18, 20, 24,$ and 28 the reliabilities of optimal graphs (0.7876, 0.9034, 0.9471, 0.9760, 0.9977 and 0.9989, respectively) are indicated also. Averages have been calculated by 30 random graphs for each m. It is clear that the average reliability of $G(16, 30)$ can be approximately achieved by optimization of $G(16, 18)$. Thus the effect of optimization has no doubt.

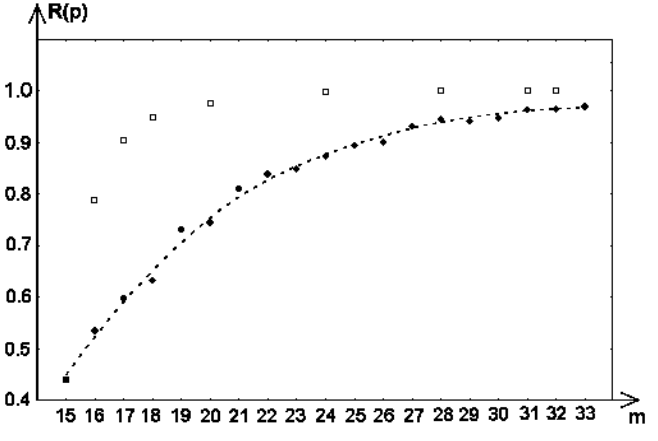


Fig. 7. Average reliability for 30 16-node random graphs in terms of the number of its edges

6 Conclusion

In this paper we have shown the use of reliability polynomials for the obtaining optimal structures for some kinds of graphs by the reliability criteria. This technique is more complex than that of optimizing by the number of covering trees (maximum) or cuts (minimum) but gives the exact results. We have shown that existence of chains in the graph structure allow to simplify the derivations significantly. Further researches can be done considering networks with unreliable nodes and reliable edges or unreliable both nodes and edges. Such networks are often better models for real communication networks but the task is quit complicated.

References

1. T. Koide, S. Shinmori and H. Ishii, "Topological optimization with a network reliability constraint," *Discrete Appl. Math., vol. 115, Issues 1-3,* pp. 135-149, 2001.
2. F.-M. Shao, L.-C. Zhao, "Optimal Design Improving a Communication Network Reliability," *Microelectronics & Reliability, vol. 37, Issue 0,* pp. 591-195, 1997.
3. J. Carlier, Li Yu and J.-L. Lutton, "Reliability Evaluation of Large Telecommunication Networks," *Discrete Appl. Math., vol. 76, Issues 1-3,* pp. 61-80, 1997.
4. N. Fard , T.-H. Lee, "Spanning Tree Approach in All-Terminal Network Reliability Expansion," *Computer Comm., vol. 24, Issue 13,* pp. 1348-1353, 2001.
5. J. Levendovszky, L. Jereb, Zs. Elek and Gy. Vesztergombi, "Adaptive Statistical Algorithms in Network Reliability Analysis," *Performance Evaluation, vol. 48, Issues 1-4,* pp. 225-206, 2000.
6. A.M. Shooman, "Algorithms for network reliability and connection availability analysis," *Electro/95 Int. Professional Erogram Proc.,* pp. 309-333, 1997.
7. B. Liu, K. Iwamura, "Topological Optimization Models for Communication Network with Multiple Reliability Goals," *Computers & Math. with Appl., vol. 39, Issues 7-8,* pp. 59-59, 2000.
8. R.-H. Jan, "Design of Reliable Networks," *Computers Ops Res., vol. 20, no. 1,* pp. 25-34, 1993.
9. O.K. Rodionova, "Application Package GRAPH-ES/3. Connectivity of the Multigraphs with Unreliable Edges (Atlas, procedures)," Preprint n. 356, Computing Center of the SB AS of the USSR, Novosibirsk, 1982. (in Russian)
10. O.K. Rodionova, A.A. Gertzeva "On the Construction of the Optimal-connected graphs", *Proc. of the ICS-NET'2001 Int. Workshop, Moscow,* pp. 200–208, 2001. (in Russian)
11. E. Ayanoglu, Cpih-Lin, "A Method of Computing the Coefficients of the Network Reliability Polynomial," *GLOBECOM '89, IEEE, vol.1,* pp. 331-337, 1989.
12. S.M. Mainagashev, M.I. Netchepurenko, "On Uniformity of Optimally Connected Multi-Graphs," *System modeling-5, Bull. of the Computing Center SB RAS,* Novosibirsk, pp. 19–24, 1979. (in Russian)
13. E.F. Moore, C.E. Shannon, "Reliable Circuits Using Less Reliable Relays," *J. Franclin Inst., 262, n. 4b,* pp. 191-208, 1956.
14. O.K. Rodionova, "Some Methods for Speed up the Calculation of Information Networks Reliability," *Proc. XXX Int. Conf. "IT in Science, Education, Telecommunications and Business," Ukraine, Gurzuf,* pp. 215-217, 2003.

Differentiated Web Service System through Kernel-Level Realtime Scheduling and Load Balancing

Myung-Sub Lee, Chang-Hyeon Park*, and Young-Ho Sohn

School of Computer Science and Electrical Engineering, Yeungnam University
Kyungsan, Kyungbuk 712-749, Republic of Korea
{skydream, park, yhshon}@yu.ac.kr

Abstract. With the rapid increase in the number of Web users and resulting development of various kinds of Web applications, Web Quality of Service(QoS) has become a critical issue for Web services, such as e-commerce, Web hosting, etc. Nonetheless, most Web servers still deal with Web user requests on a First In First Out(FIFO) basis, which cannot provide differentiated QoS. This paper presents two approaches for the differentiated Web QoS: a kernel-level approach, which adds a realtime scheduler to the operating system kernel to maintain the priority of the user requests determined by the scheduler in the Web server, and a load balancing approach, which uses IP-level masquerading and tunneling technology to improve the reliability and response speed of the Web services.

Keywords: Differentiated QoS, load balancer, masquerading, tunneling

1 Introduction

As the World Wide Web(Web) is inexpensive, easy to use, and able to provide a broad range of information, the large number of Web users significantly increases the amount of Web data, such as various kind of documents including multimedia data, transmitted through the internet[1]. Recently thus the technologies related to Web QoS(Quality of Service) which guarantees the quality of Web services are becoming more important[2,3]. Particularly for the differentiated quality of Web services, Web server must be able to classify contents depending on the importance of the information and the priority of the customer, and able to schedule the classified contents. However, most Web servers currently provide best effort services on a FIFO(First In First Out) basis only, regardless of kinds of contents. This means that, when they are overloaded, servers cannot provide the right services to the premium users[5].

In the case of the most commonly used Apache Web Server, it processes requests on a FIFO basis, although it can recognize the type of the request in the server[4]. Hence, a new server is needed that can guarantee the quality of

* Corresponding Author: park@yu.ac.kr

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 441–450, 2004.
© Springer-Verlag Berlin Heidelberg 2004

services, classify services according to specific criteria, and provide differentiated services. Despite the rapid expansion in Web use, the capacity of current Web servers is unable to satisfy the increasing demand. Consequently, even if a Web server providing differentiated services is developed, it cannot guarantee perfect service. As a resolution for Web QoS, Web server technologies employing load balancing are being proposed[13]. Load balancing Web servers are essentially guarantee service quality through the instant replies to the requests for services/connections. However, the exiting load balancing technologies for Web servers still have some problems, such as incompatiblity between different client application programs, inability to process overloaded servers, overload when processing HTTP requests/replies, packet conversion overheads, and etc.

This paper proposes two approaches for implementing load balancing Web servers that can guarantee differentiated Web QoS. In the first approach, a scheduling module is added to Web server, which assigns a priority to a client request according to its importance, and a realtime scheduler is inserted into the OS kernel so that the assigned priority is maintained in the OS, thereby providing an efficient differentiated service. In the second approach, the load balancing Web server is configured using masquerading and tunneling technologies to distribute the load by class, thereby improving the reliability and response time of the Web services.

2 Related Works

QoS attempts to guarantee a specific level of service quality, yet, in the current Internet, it is hard to guarantee QoS due to the difficulty involved in predicting which path a packet will use to reach its destination. To meet the requirements of QoS, the IETF(Internet Engineering Task Force) has proposed a number of service models and mechanisms, among which the Integrated Service/RSVP model and Differentiated Service model are actively being investigated. However, the application of the models to current networks has been delayed due to poor expansibility, the necessity for highly functional routers, and the lack of QoS functions. There has already been some research on the concept of Web QoS that guarantees a certain level of QoS to client requests by applying the concept of differentiated services to a Web server[3].

Web QoS is to classify the client requests received by the Web server according to certain classification criteria, such as the file name, user ID, and client IP, then provides a differentiated QoS according to the class. There are two main approaches to design a Web server with differentiated QoS. The first is the user-level approach, which modifies the Web server by adding a differentiation module. Yet, this approach is ineffective, as the priority assigned in the Web server is not necessarily maintained in the OS kernel, because the scheduling process in the kernel is not performed in the same way as in the Web server. The second is the kernel-level approach. Here, a differentiation module is added to both the Web server and the kernel, so that the kernel schedules in the same way as the Web server, thereby guaranteeing perfect QoS. In this paper, differentiated services are provided by adding a module for classifying and scheduling

to an Apache Web server, while adding a Montavista[15] realtime scheduler to the kernel.

Previous researches on load balancing Web servers have mainly focused on four approaches: a round-robin DNS(Domain Name Service) on the client side, round-robin DNS on the server side, scheduling on the application level, and scheduling on the IP level[13].

In the first approach, a round-robin DNS on the client side, an applet provided by the client sends a message requesting the load information of distributed servers, selects a server according to the information received from the servers, then delivers the message. Smart Client[6] developed at Berkeley University adopted this approach. However, the main weakness is that the servers are not transparent from the viewpoint of the client, so all the client applications have to be modified.

The second approach using a round-robin DNS on the server side is simple in that only the servers have to be changed. This method applies round robin to DNS so that different IP addresses are mapped in sequence, thereby distributing the load among the servers. The Scalable Web Server[7] developed at NCSA adopted this approach. Yet, when a particular server reaches overload due to client caching and the hierarchical system structure, controlling the servers can be overhead.

In the third approach, which involves application-level scheduling on the server side, as in EDDIE[8], Reverse-proxy[9], pWEB[10], and sWEB[11], a distributed server measures its own load when it receives a HTTP request, then decides whether or not to process the request. If the server is unable to process the request, it forwards the request to another server, obtains the result, then finally transmits the result to the client. However, this approach involves a transmission delay due to two or more TCP connections, and can occur a heavy overhead on the application level in relation to processing HTTP requests and replies.

In the last approach, which involves IP-level scheduling on the server side, as in the Magic Router[12] developed at Berkeley University and Local Director[14] developed by Cisco, network address translation(NAT) is used to make several concurrent services at different servers appear to be services from a single IP address. The NAT-based system is generally composed of a load balancer that performs scheduling to distribute the load, and real servers that provide actual Web services. In the NAT method, if the number of real servers exceeds 20, a bottleneck occurs in the load balancer. In addition, the packet rewriting overhead is high.

In this paper, we propose a load-balancing Web server that resolves the packet rewriting overhead and bottleneck in the load balancer due to IP-level scheduling by combining a network address translation technique and IP tunneling technique. In addition, the proposed server investigates the number of connections according to the service class using a DLC(Differentiated Least Connection) algorithm to improve the weakness of the existing LC algorithm[13].

3 A Differentiated Web Service System

The whole structure of the differentiated Web service system proposed in this paper is shown in Fig. 1 and detailed explanations are given in the following sections. The proposed system uses two approaches: kernel-level approach and load-balancing approach.

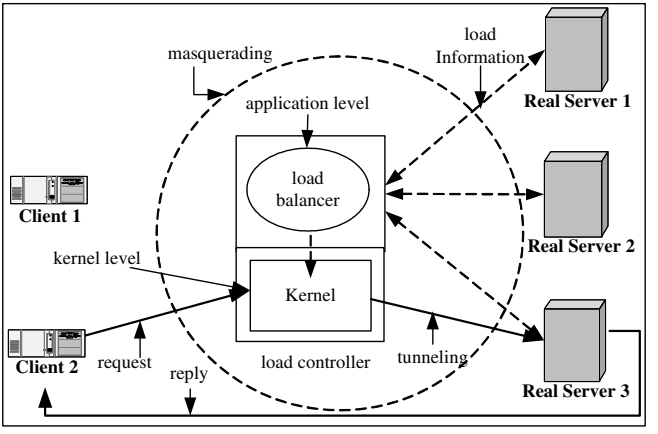


Fig. 1. Structure of differentiated Web service system

3.1 Kernel-Level Approach

For the client requests, the kernel-level approach maintains their priority order determined by the Web server in the OS kernel. This approach is implemented

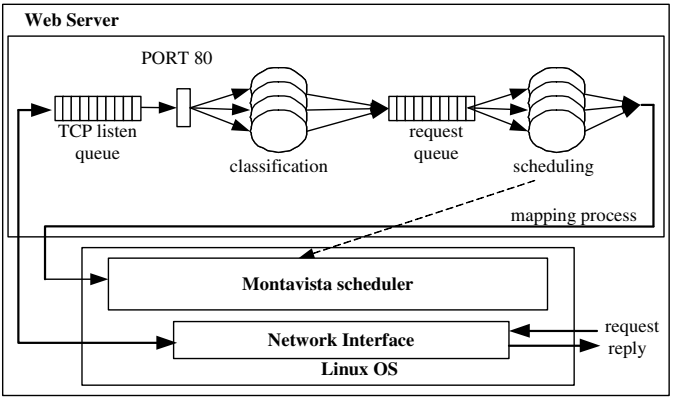


Fig. 2. Process mapping in kernel-level approach

by mapping the scheduling processes in the Apache Web server to the realtime scheduling processes in the OS kernel. Fig. 2 shows the process mapping between the scheduler in the Web server and the scheduler in the kernel.

As shown in Fig. 2, when the client requests come through a Network Interface Card(NIC), the Web server receives them from port 80 in the TCP listening buffer, classifies them by connection according to specific classification policies(client IP, URL, file name, directory, user authentication, etc.), assigns the proper priority, then inserts them into the corresponding queues. Thereafter, at the same time the requests are being scheduled, the scheduling processes in the Web server are mapped one-to-one to the processes in the realtime scheduler(Montavista in this paper) in the Linux OS kernel.

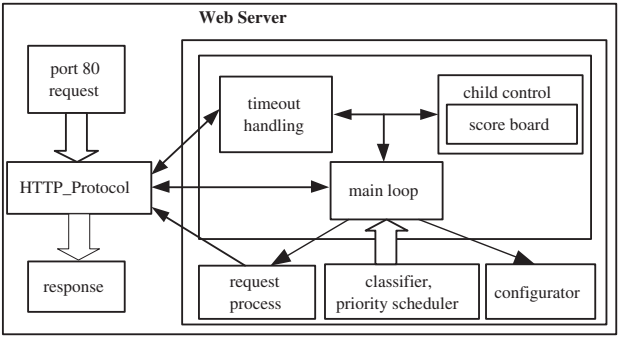


Fig. 3. Diagram of modified Web server configuration

Fig. 3 shows a diagram of the modified Web server configuration, where a master process is created when the modified Apache Web server starts. The process generates child processes for a prime-level class, high-level class, and default class, then reconfigures the http daemon to reflect this. The requests classified as prime-level and high-level are scheduled by the realtime scheduler, while the others are scheduled by the original kernel scheduler.

3.2 Load Balancing Approach

The load balancing Web server proposed in this paper has a high performance and expansibility by enhancing the packet transmission rate and by resolving the bottleneck in the load balancer through the use of IP-level masquerading and tunneling. In the proposed system, a single load-balancer distributes the requests to several real servers, which share a common IP address, using a masquerading technique so that they look like a single server from the outside. The load balancer of this paper is composed of a kernel-level part and IP-level part.

Kernel-Level Part. IP masquerading hides the real servers behind a virtual server that acts as a gateway to external networks. Fig. 4 shows the structure of

the kernel-level part of the load balancer, where the clients send their requests using a real IP(e.g. 165.229.193.10), then the mask-gate rewrites the connection information in the packets before delivering them to the internal network. As such, the clients can communicate with servers without knowing their connection information. The tunneling technique performs encapsulation, which adds the virtual IP address of each server to the header of the IP packets with a publicized real IP address, and decapsulation, which is the reverse process of encapsulation. As a result of tunneling, servers receiving packets can send data directly to external networks using the IP address in the request packets without having to rewrite the address.

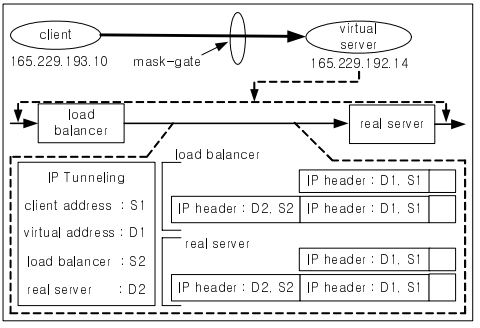


Fig. 4. Kernel-level part of load balancer

IP-Level Part. The IP-level part of load balancer is composed of contents extraction module, classification module, and DLC algorithm, as shown in Fig. 5. When the clients send requests, the contents extraction module and the classification module in the IP layer perform classification and scheduling. To handle request messages received by the IP layer, `sk_buff`, a data structure of the Linux kernel, is used to obtain the path of the HTTP request data, while the contents extraction module extracts the HTTP request packets from the client request packets.

Contents extraction is carried out in response to the HTTP request packets among the client request packets and involves the following steps:

1. Extract TCP and UDP values using structure `sk_buff`, and assign a value to the protocol variable.
2. If th protocol variable stands for TCP value, extract the incoming packets through port 80(processes HTTP requests only).
3. If the incoming packets have data, extract the file names from the URI.
4. After moving the data offset to next packet, store the URI to an array filename.

Meanwhile, the contents classification module classifies contents by comparing the extracted file names to the data structure for classes of class names, which uses the following steps.

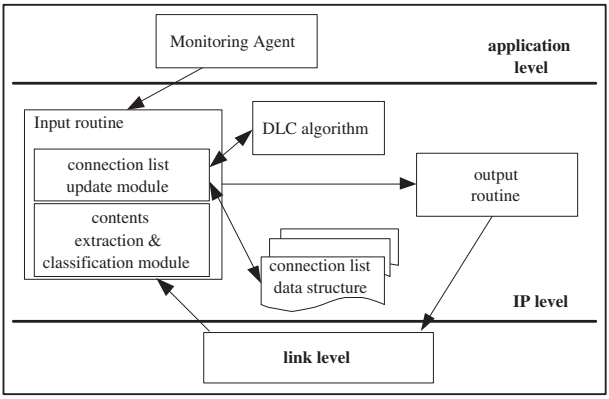


Fig. 5. IP-level part of load balancer

1. Classify file names requested by clients into classes.
2. Extract data structure and file names that include information about the file priority and compare them with the classified information.
3. If corresponding information is found, return the priority of the file name.
4. If no corresponding information is found, assign the lowest priority.
5. For a variable with an assigned priority, transmit the scheduling result and the packet, and update the number of connections.

To schedule connections of the load balancer, this paper proposes a DLC algorithm, a connection-based scheduling algorithm, which is modified from LC algorithm[13] to consider the connections by classes. The DLC algorithm involves the following steps.

1. Receive classified class information in the form of a linked list, scan the list from the head and count the number of connections to each real server.
2. Count the number of connections to each real server by classified class from the top class to the current class(carry out the same counting on classes excluded in the classification).
3. Compute the ratio of the count to the total number of connections to each real server and perform scheduling based on the ratio.
4. Return a real server with the least number of connections.

4 Implementation and Experiment

The differentiated Web service system proposed in this paper is implemented using a Linux Kernel 2.4.7 and PCs with a Pentium-III 800MHz processor and 256MB RAM, while the test environment is built by networking three clients, one load balancer, two servers, and one monitoring server. An Apache Web Server 2.4.17 is modified for the Web server, and a Montavista realtime scheduler is added to the Linux kernel.

4.1 Client interface

The client interface is a GUI that tests the performance of a particular Web server by sending a request for a Web page in the server based on a certain time unit and transmission rate, within which a client can send requests to real servers via the load balancer. Fig. 6 shows that the client interface displays the reply changes of servers. The bottom left window presents the test environment settings and contains the "server address", which is the virtual server address to which a request is sent, the "total number of sessions", which is the total number of connections, the "concurrent users for each session", which is the maximum number of users that can connect concurrently, and the "number of calls per session", which is the number of sessions requested per connection. The bottom right window presents the realtime test results in text, including the reply counts and reply rates per time unit. The top window presents the linear trends of the reply counts and reply rates over time, where the numbers in the leftmost column are the reply counts, and those in the rightmost column are the reply rates.

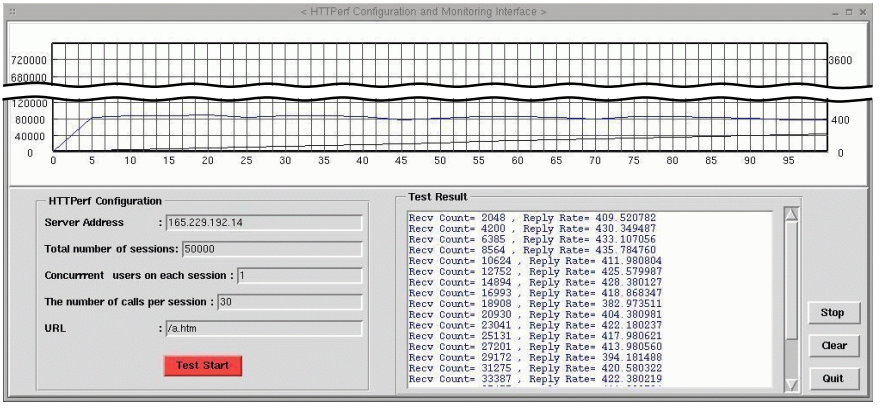


Fig. 6. Client interface

4.2 Experiment

Tests are carried out for three cases: when the servers are not overloaded(test 1), when the servers are overloaded(test 2), and when the servers are overloaded and some requests are subsequently stopped(test 3). In test 1, the virtual IP address is 165.229.192.14, the total number of connections 50000, the number of concurrent users per session 1, and the number of calls per session 50.

Fig. 7(A) presents the results of the client interface, which shows the reply changes of Web servers upon the three clients. As the servers are not overloaded, the graphs are almost the same. In this situation, differentiated services are

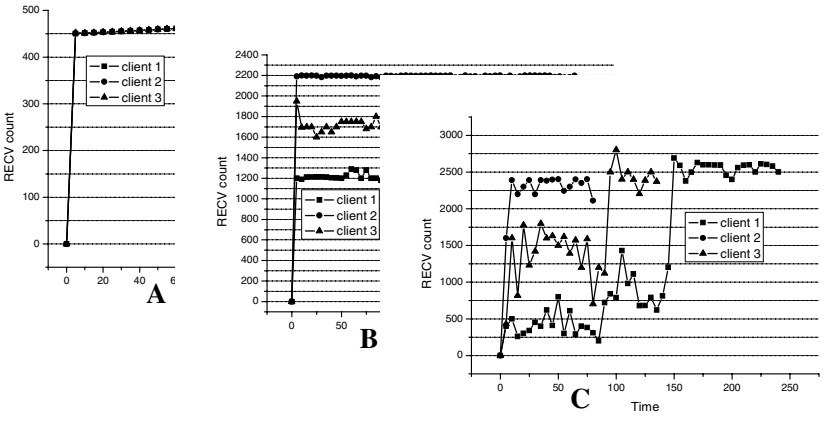


Fig. 7. Experimental graphs of real servers

not necessary, as the three classes are all well served. That is, if Web servers are not overloaded, all classes of request are processed smoothly, so there is no problem in the Web services. However, if Web servers are overloaded, high priority requests may not be served properly. Thus, in test 2, the virtual IP address is 165.229.192.14, the total number of connections 50000, the number of concurrent users per session 30, and the number of calls per session 50. In Fig. 7(B), the top line shows the reply rate for client 2 requesting a.html, the middle line shows the reply rate for client 3 requesting b.html, and the bottom line shows the reply rate for client 1 requesting c.html. As priority is assigned in the order of a.html, b.html, and c.html, the reply rate for client 2 is the highest, while that for client 1 is the lowest. In test 3, which uses the same conditions as test 2, the requests from all classes continue to occur for 80 seconds, then the request for a.html stops. As shown in Fig. 7(C), the reply rates for the requests for b.html and c.html increase, and the request for b.html have the highest priority. Likewise, when the request for b.html is stopped after 130 seconds, the reply rate for the request for c.html increases.

5 Conclusion

To implement a differentiated Web service system that provides differentiated services according to information importance or user priority, this paper proposed two approaches: a kernel-level approach and a load-balancing approach. In the kernel-level approach, a realtime scheduler is added to the kernel, while in the load-balancing approach, the load balancer is implemented using an IP-level masquerading technique and tunneling technique. For the load balancer, a new DLC algorithm was proposed that improves on the existing LC algorithm by providing differentiated Web services according to the priority of the service re-

quest. The performance of the load balancing system was tested in three different situations, and the results confirmed that the system supported differentiated Web services. Like the LC algorithm, the proposed DLC algorithm works statically and thus is unable to reflect dynamic load changes in each real server. Hence, to resolve this problem, further research on a dynamic load balancing service system that reflects the degree of load on servers through server CPU monitoring and server state analysis is currently in progress.

References

1. R.Fielding, J. Getys, J. Mogul, H. Frystyk, and T. Berners-Lee, Hypertext Transfer Protocol HTTP/1.1, IETF (1997)
2. N. Bhatti, A. Bouch, and A. Kuchinsky, "Integrating User Perceived Quality into Web Server Design", Proc. of the 9th International World Wide Web Conference, Amsterdam, Netherlands (2000) 92-115
3. N. Vasiliou and H. Lutfiyya., "Providing a Differentiated Quality of Service in a World Wide Web Server", Proc. of the Performance and Architecture of Web Servers Workshop, Santa Clara, California USA (2000) 14-20
4. Apache Group, http://www.apache.org/.
5. R. Bhatti and R. Friedrich, "Web Server Support for Tiered Services", IEEE Network (1999) 64-71
6. Chad Yoshikawa, Brent Chun, Paul Eastharn, Armin Vahdat, Thomas Anderson, and David Culler, "Using Smart Clients to Build Scalable Services", USENIX'97, http://now.cs.berkeley.edu/ (1997)
7. Thomas T. Kwan, Robert E. McGrath, and Daniel A. Reed, "NCSA's World Wide Web Server: Design and Performance", IEEE Computer (1995) 68-74
8. A. Dahlin, M. Froberg, J. Walerud and P. Winroth, "EDDIE: A Robust and Scalable Internet Server", http://www.eddieware.org/ (1998)
9. Ralf S.Engelschall, "Load Balancing Your Web Site: Practical Approaches for Distributing HTTP Traffic", Web Techniques Magazine **3** http://www.webtechniques.com (1998)
10. Edward Walker, "pWEB - A Parallel Web Server Harness", http://www.ihpc.nus.edu.sg/STAFF/edward/pweb.html (1997)
11. Daniel Andresen, Tao Yang, Oscar H. Ibarra, "Towards a Scalable Distributed WWW Server on Workstation Clusters", Proc. of 10th IEEE Intl. Symp. of Parallel Processing(IPPS'96) (1996) 850-856
12. Eric Anderson, Dave Patterson, and Eric Brewer, "The Magicrouter: an Application of Fast Packet Interposing", http://www.cs.berkeley.edu/∼eanders/magicrouter/ (1996)
13. Wensong Zhang, "Linux Virtual Server Project", http://proxy.iinchina.net/∼wensong/ippfvs (1998)
14. Cisco System, "Cisco Local Director", http://www.cisco,com/warp/public/751/lodir/index.html (1998)
15. Montavista Software, http://www.montavista.com/.

Adaptive CBT/Anycast Routing Algorithm for Multimedia Traffic Overload

Kwnag-Jae Lee[1], Won-Hyuck Choi[1*], Jung-Sun Kim[2]

[1] School of Electronics, Electronics and Multimedia, Seonam University, 702, Kwangchi-dong, Namwon-city, Jonbuk, 590-711, Korea, kjlee@seonam.ac.kr,rbooo@korea.com

[2] School of Electronics, Telecommunication and Computer Engineering, Hankuk Aviation University, 200-1, Hwajeon-dong, Deokyang-gu, Koyang-city, Kyonggi-do, 412-791, Korea jskim@mail.hangkong.ac.kr

Abstract. According to the social demand to the expansion of internet use and construction of ultrahigh speed network, the realtime and the maximized capacity multimedia data service gets exceeding the infra level of communication. As a response to the communication condition, multicast service provides effective use from various resources and corresponds actively to high-speed data transmission. For these reasons, multicast service is considered as a major internet solution for the next generation. Thus in this study, various multicast routing methods are proposed and an analysis for CBT routing method based on CBT routing protocol from the existing multicast routing protocol is made. As a result, there happens traffic congestion to a core router because of structure problem of CBT protocol, thus performance of the whole routing declines as a consequence. The AIMD (Additive Increase Multiple Decrease) algorism applied Anycast routing method that is suitable for traffic dispersion in high bandwidth according to increment of traffic load from CBT Shared Tree Routing Method is strongly suggested in this thesis.

1 Introduction

Multicast protocol classifies network users into specific groups and provides not only various but characterized services with communicating protocol to individuals, enterprises, and the government. It becomes a matter of concern and interest for internet communication. As a refection of this demand, development of multicast service, research for efficiency improvement, and are actively in process, and recently various protocols for quality improvement and reliable transmission have been proposed. In multicasting protocol, it uses shortest path tree from itself and the representative protocol are SBT (Source Based Tree) method that connects a gap between a transmitter and a recipient through shortest path and ShT (shared Tree), the covalent tree method, that one network router becomes a center and it sets the shortest path then transmits data packet from the recipient to each members.

* The corresponding author will reply to any question and problem from this paper

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 451–459, 2004.
© Springer-Verlag Berlin Heidelberg 2004

The CBT (Core Base Tree) method, the representative protocol of the covalent tree, is one of methods to improve high-speed transmission of multicast packet and efficiency of communication by decreasing overhead from tree constitution's overlap. However, CBT (Core Base Tree) has several problems in structure and they work as its vulnerability (Core Tree Base).

The first problem of CBT is the phenomenon of transmitter's traffic concentration around Core Router. For instance, traffic density and surplus sign around Core router that are often seen in services like video, Telnet, Ftp, etc. The figure 1 shows concentration problem in traffic and the figure 2 is Poor Core phenomenon. The core's ideal position in traffic reception is right in the middle that correspondent with the size of distance from group members.

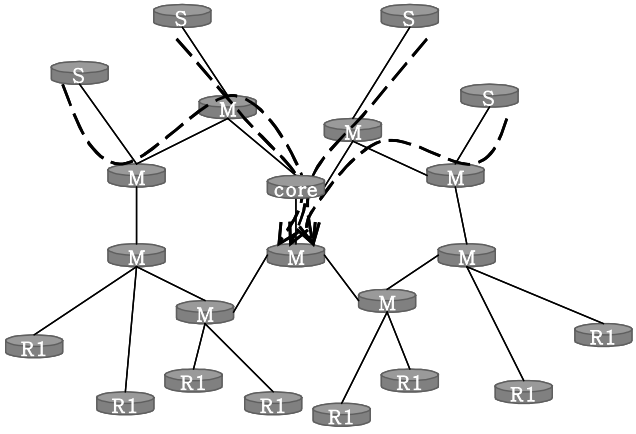


Fig. 1. Traffic Concentration

However, if the core is positioned in an isolated area from transmitter-recipient of packet and used independently, then it becomes impossible to have right choice and practice even though it does not require much the high bandwidth and the maintenance space of routing information. Therefore, ABT (Anycast Based Tree) is proposed in the thesis.

ABT does not limit core in specific position within network but let it actively be located so that the previously mentioned problems of CBT can be solved. The specific resolution is to use AIMD (Addictive Increase Multiple Decrease) algorism. The controlled transmission rate of traffic enables traffic that is concentrated in core router, to maintain average transmission rate and leads traffic to poor core so it helps to improve excess use in whole system and performs multicast service in high speed.

In the thesis, several topics were addressed to compare and analyze efficiency of each multicasting routing method; the characteristics of CBT is in chapter 2, theory and effect of ABT in chapter 3, and amount of transmission and measurement of transmission delay for CBT and ABT in chapter 4.

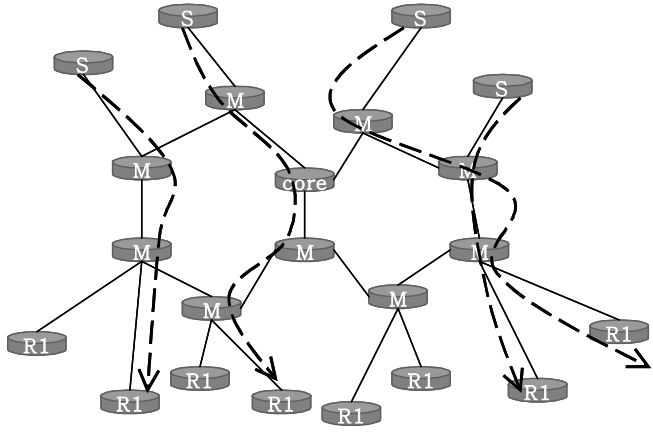


Fig. 2. Poor core placement

2 CBT

The existing methods of multicast protocol are classified to source based SBT (Source Based Tree) multicast protocol that consists separate trees in each source and ShT (Shared Tree) multicast protocol, which multi sources share tree in the system[5].

The symbol $(*, G)$ indicates ShT (Shared Tree). The actual size of tree is shared tree so it has result of $O(|G|)$ regardless of number of source. The cost according to the constitution of tree may cause a serious traffic delay if there is increment of economy or number of source. ShT is appropriate when it needs to deal with traffic of relatively small bandwidth and to apply multicast service in network that has multiple transmitters.

There are several methods based on shared tree. The first one is CBT (Core Based Tree) protocol method and BT routing tree takes care protocol by placing a core router in the center of shared tree.

PIM-SM protocol has a barrier to choose the optimal routing root because it is operated in one-way tree; however, CBT tree is operated in two-way tree so its network expansion is superior than the existing method for source based multicast routing.

PIM-SM protocol uses RP (Rendezvous Point) for the each multicast group, which every routing reception shares, and it is used in one-way tree.

3 ABT (Anycast-Bast-Tree)

3.1 Problem

Anycast provides structural resolutions of CBT, such as concentration of traffic to core, Poor Core phenomenon that is caused by the failure of using strategy like setting core position, etc. can be observed in CBT tree.

In other words, ABT deals with traffic concentration in core when the core of groups is in mapping and also performs multicast by constituting Non-Core tree that has smaller bandwidth than the core router but has excellent ability in multicast process, in order to prevent unnecessary use of bandwidth of mapping available core router that is away from the groups.

This tree assigns the entry of Anycast tree to a near router that is via a core, and constitutes system to transmit multicast data packet directly from the router to each members in need. Thus through these process, ABT can disperse traffic concentration in core, use every router in network effectively, and finally enhance speed of process by appropriate decentralizing of data. Such decentralizing in traffic also decreases packet delay that is often seen in core router; therefore this enhancement can lead better quality in service and the network's effective operation can prevent Poor Core from the beginning.

3.2 ABT Rate-Base Control

The main characteristic of the suggested ABT is its treatment of multicast packet in a formation like CBT without having core router. In this process, however, it requires control mechanism that moves traffic to core in the other side when traffic gets concentrated in core more than threshold.

In multicast routing, the time for traffic to pass the core is called Core round trip time (crtt), and crtt becomes reset time for a table in a transmitter and a control parameter.

For increase factor of packet, the transmission time increment of core can be shown as $\frac{a}{crtt}$, and if there is increase in transmitter, the formula is like below.

$$Rin=Rnow+\frac{a}{crtt} \quad (1)$$

Here, R*in,* is transmit packet and R*now* is amount of packet in the present core.

If there is decrease of packet to core, the formula becomes like below and b is factor for decrease.

$$Rin=\frac{Rnow}{b} \quad (2)$$

The increase of transmitted packet and the average transmitted rate based on decrease can be calculated at the core from *a* and *b*. Also the transmission rate is calculated according to the size of packet from recipient and the minimum and maximum rate for transmission can be calculated with increase of recipient.

R*min* is minimum rate for transmission of core, R*max* is maximum rate for transmission of core, and transmission time increase is n.

$$Rmax=Rin\frac{a}{crtt}n \ (3)$$

$$Rmin=\frac{Rmax}{b} \ (4)$$

The below formula is for the average rate for transmission by using min. and max. rate for transmission of core.

$$Rave=\frac{1}{2}(Rmax+Rmin)=\frac{a}{2}\bullet\frac{b+1}{b-1}\bullet\frac{n}{crtt} \ (5)$$

According to the formula (5), Poor Core phenomenon that becomes the minimum rate for transmission of core and congestion around core that occurs it becomes the maximum rate of transmission can be controlled with the average rate of transmission.

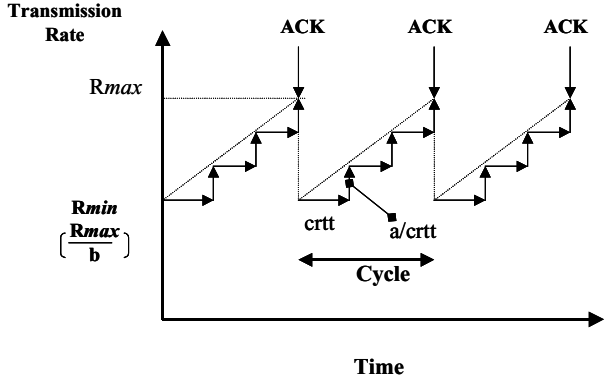


Fig. 3. The rate of transmission based on increase in transmission

The figure 3 shows retransmission of ACK at the end of each cycle after the multicast packet (n, crtt, Rave) completed transmission. The formula for the average rate for transmission in core router is like below. At this time, loss of packet through retransmission of ACK is *p*.

$$Tran=\frac{1}{crtt}\sqrt{\frac{a}{2}\bullet\frac{b+1}{b-1}\bullet\frac{1}{p}} \ (6)$$

3.3 Operation of ABT

The domain of tree administration in general multicast tree can be classified into a Join process to ask participation for group member of a host and a Prune process to eliminate branch of tree according to group membership state. Especially, Prune is processed in either Down-Stream or Up-Stream direction.

Similar to the general shared tree, Down-Stream Prune is not the Root Router that is a starting point of tree but a way of process when the connected child nods to router is no longer group member.

3.3.1 Tree Join of Anycast

I. Any host that wants to join to the tree of Anycast transmits multicast to JOIN_REQUEST message and every link that is connected to itself with group and associated Anycast address.
II. The message received local router invokes Joining process to connect to Anycast tree. In this process, the local router does not confirm Anycast entry in the routing table and only initializes the received group address of Anycast.
III. Later the local router relays JOIN_REQUEST message to the next Hop router on the root, which is on its way to Anycast group. JOIN_REQUEST message is finally transmitted to On-Tree router, and the message registered router transmits the JOIN_ACK through backward root of JOIN_REQUEST message with Down-Stream. Like in Up-Stream, each router initializes timer by using Tree-Flag to maintain temporary condition of assigned group with JOIN_ACK message and when time elapse router manages the timer as a member of Anycast.
IV. Once JOIN_ACK message reaches to router, a new recipient shows the new branch does not form any loop to prove Anycast tree is Loop-Free (under an assumption that the present tree does not involve loop), and through this process On-Tree router of Anycast expands new branch of tree as it conducts its own duty like a core in each CBT tree.

3.3.2 Tree Prune of Anycast

I. The process of change in root as a Child can be made by giving up a member for Anycast tree and transmitting ROOT_QUIT massage to a Child.
II. Once a Child received the message, it transmits ROOT_QUIT_ACK message and then concludes by its proclamation of being root. Such prune process of tree provides pliability in tree constitution within one domain and can actively reflect reconstitution of tree topology according to any change in members.
III. Anycast tree is maintained through confirmation of ECHO message like CBT method, and if there is no reply message of ECHO it operates tree by using FLUSH message.

4 Simulation Model and Evaluation

4.1 Simulation Topology

For CBT routing and application of Anycast that are introduced in the chapter 3, the congestion that is formed around CBT core is examined and analyzed with traffic rate. Based on theoretical approach, simulations that convert to Anycast routing according to the stream of traffic in CBT routing method and the state of Core link are executes.

The figure 4 indicates a set of applied simulation topology. While CBT multicast routing is executing, queueing model is analyzed. Also CBCS (Core Bottleneck Calculation Server) is set to the core router, and it is executed with initialization of CBT Tree.

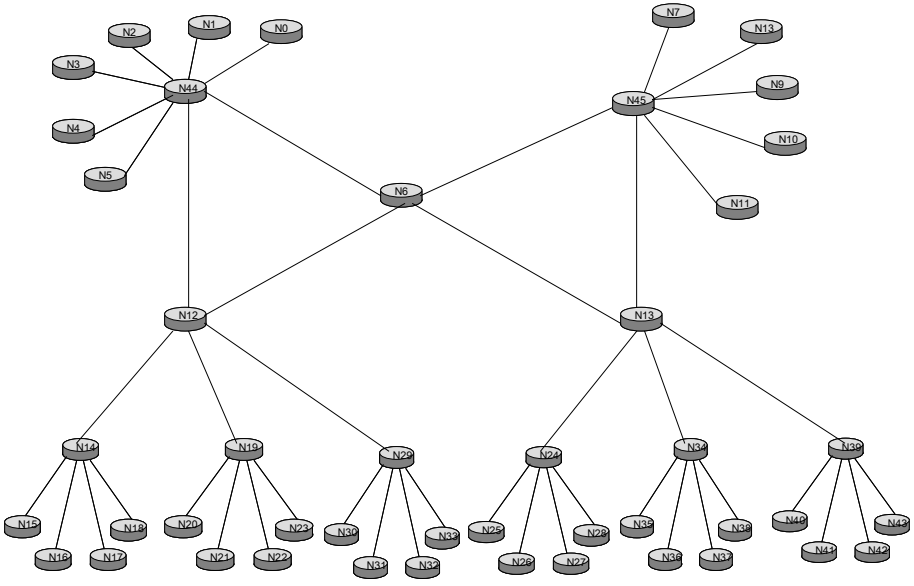


Fig. 4. Simulation Topology

The exterior condition of the simulation are; memory 512 MByte capacity, PC that uses Intel Pentium 4 CPU of system clock 1.5 GHz as a platform, operating system is Linux Redhat 7.0, and simulation device is ns-2(Network Simulator Version 2) that is widely used as a simulator PC based condition.

For efficiency of the proposed multicast network routing evaluation, two different categories are carefully examined; The congestion of CBT Core in network topology and the congestion in Anycast. With consideration of characteristics of multicast, each data packet are classified to 512, and 1024 Byte then analyzed.

4.2 Comparison and Measurement of CBT and Anycast

To the simulation model, each CBT routing protocol is applied and the numbers of multicast groups and transmitters are varied. Then the packet process condition of Core router is measured based on the sized of multicast data packet. The figure 5 and 6 show the result of the simulation.

It is clear that characteristic of queueing delay is somewhat superior when the sized of data packet is small like 512 Byte. However, there is abrupt queueing delay when the sized of data packet is big like 1024Byte as it starts to deal with 20 packets in a second, and it causes congestion in core.

From the result, we confirm the characteristic of queueing delay is somewhat superior when size of data packet is small like 512Byte. However, there is abrupt queueing delay and causes congestion as the system deals with 20 packets in a second when size of data packet is large like 1024 Byte.

The reason for this is that there is formation of initialization of multicast tree and is frequent *Join* and *Leave* of group, thus interval for packet's arrival becomes shorter and relatively increase in load of packet occurs as a result.

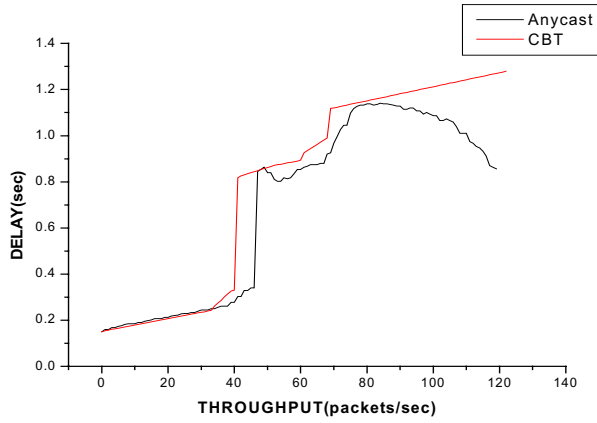


Fig. 5. 512Byte Packet transmission delay of CBT/Antcast Core

The figure 5 shows 512Byte packet transmission delay of CBT/Anycast and the figure 6 shows 1024 Byte queueing delay of core as the system is executed as Anycast routing protocol in CBT routing protocol. There is difference in interval to have congestion depends on size of multicast packet, however it is obvious there happens queueing delay of core router in a same shape because of increased packet from multicast group**'s** Join. Since every group Joins to multicast tree and renews Routing Table, there is conversion of routing protocol to Anycast method even in congestion with abrupt queueing delay, and such conversion to Anycast method slowly but definitely decreases queueing delay.

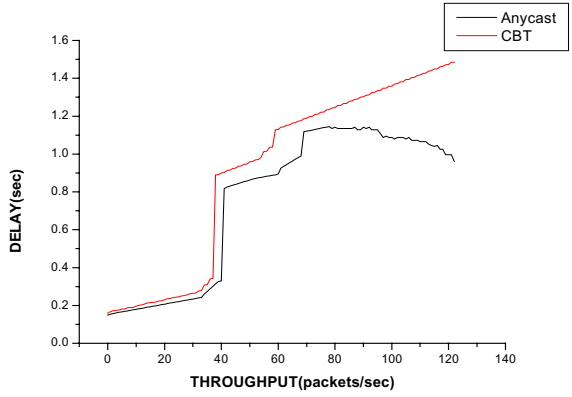


Fig. 6. 1024 Byte Packet transmission delay of CBT/Anycast core

5 Conclusion

In this thesis, change of routing methods from CBT shared tree routing method that is stable in relatively low bandwidth to Anycast routing method that is appropriate for traffic congestion even in high bandwidth depends on load of traffic. In the study, characteristics of delay according to the size change of multicast data packet when the system is changed from CBT to Anycast, were observed and evaluated.

For multicast tree, if there is a service of multimedia data that demands big bandwidth while using CBT routing method in relatively small bandwidth demanding traffic condition, Link condition of CBT core should considered. Then, the strategy of CBT/Anycast routing method, which converts to Anycast routing, will be used to enhance the efficiency of multicast protocol.

Referenece

1. M. Parsa and J. J. Garcia-Luna-Aceves, “A protocol for scalable loop-tree multicast routing,” IEE IEEE J. Select. Areas Commun., vol. 15, pp. 316_331, Apr. 199 optim-07.txt, November, 1997.
2. X. Jia, and L. Wang, "A Group Multicast Routing Algorithm by using Multiple Minimum Steiner Trees", Computer Communications, pp.750 -758, 1997.
3. A. Ballardie, "Core Based Trees (CBT) Multicast Routing Architecture", RFC2201, 1997.
4. A. Ballardie, "Core Based Trees (CBT Version 2) Multicast Routing Protocol Specificastion RFC2189, 1997.
5. J.Moy, “Multicast Extensions to OSPF”, IETF RFC 1584, 1994.
6. K. Ettikan, "An Analysis Of Anycast Architecture And Transport Layer Problems", Asia Pacific Regional Internet Conference on Operational Technologies, Kuala Lu- mpur, Mal- aysia, Feb.,-March, 2001.
7. J. Lin and S. Paul, "RMTP: A Reliable Multicast Transport Protocol," IEEE INFOCOM '96, San Francisco, CA, March 1996.R. Yavatkar, J. Griffioen, and M. Sudan, "Reliable Dissemination
8. W. Yoon, D. Lee, H.Youn, S. Lee, S. Koh, "A Combined Group/Tree Approach for Many-to-Many Reliable Multicast," IEEE INFOCOM'02, June 2002
9. B. N. Levine, S. Paul, J. J. Garcia-Luna-Aceves, "Organizing multicast receivers deterministically by packet-loss correlation," the sixth ACM international conference on Multimedia, pp201-210, September 1998
10. "The Network Simulator: ns-2," http://www.isi.edu/nsnam/ns/

Achieving Fair New Call CAC for Heterogeneous Services in Wireless Networks

SungKee Noh[1], YoungHa Hwang[1], KiIl Kim[2], and SangHa Kim[2]

[1] Electronics and Telecommunications Research Institute
{sknoh, hyh}@etri.re.kr
[2] Department of Computer Science, Chungnam National University
{kikim, shkim}@cclab.cnu.ac.kr

Abstract. In wireless mobile network, handoff dropping probability and new call blocking probability are main issues for satisfaction of quality of service (QoS) requirements. Seeking just to control handoff dropping probability and to optimize utilization without any consideration of fair allocation, serious unfairness occurs among new connections with different QoS requirements. In this paper, we propose a novel call admission control (CAC) scheme and resource management algorithm that guarantee both short-term and long-term fairness between heterogeneous services with different traffic properties and enhance resource utilization of system. These improvements are largely dependent of seeking reservation partition for each class based on stochastic control. We analyze the system model of a cell using two-dimensional Markov chain and Neut's matrix-geometric solutions. By numerical analysis, we demonstrate that our CAC scheme actually achieves call blocking probability (CBP) fairness for wideband and narrowband calls and improve resource utilization.

1 Introduction

To meet QoS requirements in wireless network, call dropping probability (CDP) in addition to call blocking probability (CBP) must be controlled not exceeding desired QoS requirements. To achieve these basic requirements, CAC scheme becomes a problem of utmost importance. Especially, in case of QoS for multi-class traffic with different properties, it incurs much more complex problem than case of single-class [1-3]. In order to guarantee QoS of heterogeneous traffic, various approaches [4-8] have been studied. Their major objective is to develop efficient method to maximize network utilization while keeping CDP of multi-class below QoS profile. However, in most schemes, the wideband calls are hardly admitted so that serious CBP unfairness occurs. Thus, a new CAC algorithm has been recently developed to admit all type of services fairly.

To overcome serious CBP unfairness between wideband and narrowband calls in wireless networks, Epstein et al. [9] suggested fair CAC algorithm via blocking probability measurement function (BPMF), which enables to control relative admitting probability between wideband and narrowband calls. Such BPMF algorithm serves to

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 460–470, 2004.
© Springer-Verlag Berlin Heidelberg 2004

block users of an "overprivileged" class in order to accommodate users of "underprivileged" classes. To achieve this, independent multiclass one-step prediction-complete sharing and reservation (IMOSP-CS and IMOSP-RES) incorporates with a new resource management, which partitions the available bandwidth to reflect the desired blocking probability profile. The much bandwidth is allocated to underprivileged calls if CBP ratio between services is greater than the predetermined threshold. The numerical results demonstrate that BPMF actually achieve CBP fairness between wideband and narrowband calls.

But, IMOSP controls the reservation partition by simple resource management algorithm so that it often leads to system abnormalities depending on traffic behavior. Above all, IMOSP cannot guarantee short-term fairness in normal traffic conditions, much less cannot guarantee long-term fairness under heavy traffic conditions. To cope with it's weakness, we develop new CAC algorithm and resource management using biased coin method [10-13] that guarantee both short-term fairness and long-term fairness, as well as improve resource utilization.

In this paper, we propose a novel CAC scheme to admit wideband and narrowband calls fairly. In addition to CAC scheme, a new resource management algorithm is developed not only to prevent system abnormality, but also improve resource utilization. In proposed algorithm, reservation partitions for each class are dynamically adjusted while keeping CBP of each class similar or equal. The proposed method is analyzed using a two-dimensional Markov chain and Neut's matrix-geometric solutions [14-15].

The reminders of this paper are organized as follows. We discuss unfairness problem on previous researches in introduction. A novel CAC algorithm is described in section 2 and traffic model and queuing analysis of the models is described in section 3. The section 4 describes dynamic bandwidth allocations. The comparative numerical analysis is explained in section 5. Finally, we make a conclusion.

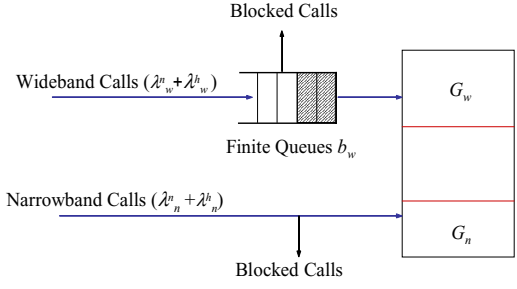


Fig. 1. System model for a cell

2 Fair Call Admission Control Policy

For each cell k, the base station architecture is illustrated in Fig. 1. Channels are divided by three sub-channels. Designed G_w and G_n channels are dedicated for wideband traffic and narrowband traffic, respectively. The shared channels can be used by either type of traffic. Narrowband calls are blocked if permitted channels are all used. But, wideband calls have finite queues so as to keep a certain amount when permitted channels are all busy. In most cases, the wideband calls are hardly admitted so that serious CBP unfairness occurs. Thus, we allocate finite buffers for wideband traffic to admit all type of services fairly.

When a new user arrives in a cell, the proposed CAC algorithm decides acceptance or rejection based on each call's current resource occupancy, reservation partition, and dynamic guide channels. A new narrowband call is admitted if the number of existing narrowband calls is less than the number of guard channels G_n for narrowband traffic. When the number of existing narrowband calls is greater than or equal to the number of guard channels G_n for narrowband traffic, a new narrowband call is accepted when the total existing used channels are less than a predefined threshold. Hand-off narrowband calls are accepted as long as the channels are not full. A new wideband call is accepted if the buffer is not exceeding a predetermined threshold. Hand-off wideband calls are accepted as long as the buffer is available.

```
if narrow_new_call is requested then
   if narrow_new_call is less than Gn
      Accept
   else if (existing used channels < Tn) then
      Accept
   else
      Reject

if narrow_handoff_call is requested then
   if (existing used channels < C) then
      Accept
   else
      Reject
```

3 Traffic Model and Analysis

In our work, we assume that a system is shared by two traffic classes, wideband and narrowband calls. The wideband call requires m bandwidths. The narrowband call requires one basic bandwidth. It is assumed that the new call and hand-off calls are arrived according to a Poisson process with mean arrival rate λ^n_n, λ^n_w and λ^h_n, λ^h_w, respectively and that service time is exponentially distributed with mean service time of $1/\mu_{ns}$ and $1/\mu_{ws.}$ Furthermore, the time that calls stay in the cell before moving into other cells also follows an exponentially distribution with mean $1/h_n$ and $1/h_{w.}$ We also describes that narrowband and wideband calls are Poisson distributed with arrival

rates λ_n ($\lambda^n_n+\lambda^h_n$) and λ_w ($\lambda^n_w+\lambda^h_w$), respectively, Moreover channel occupancy times for narrowband and wideband calls are summed with means $1/\mu_n$ ($1/(\mu_{ns}+h_n)$) and $1/\mu_w$ ($1/(\mu_{ws}+h_w)$), respectively. We allocate finite buffers B_w for wideband traffic. Let C be the total number of channels and G_n and G_w be the dedicated channels for narrowband and wideband traffic, respectively.

Then, the system can be modeled as a two dimensional Markov process, characterized by $\{i,j\}$, where i and j are the numbers of narrowband and wideband calls in the system, respectively, and the state space is represented by the set $\{s(i,j) \mid 0 \le i < G_n,\ 0 \le j \le \lfloor (C- G_n+ B_w)/m \rfloor$ and $G_n \le i \le C- G_w,\ 0 \le j \le \lfloor (C- i+ B_w)/m \rfloor\}$. $\lfloor x \rfloor$ denotes the greatest integer smaller than or equal to x. Also, let the steady-state probability that the system is in state $s(i, j)$ be $p(i, j)$. The steady-state probability vector $\boldsymbol{p}$ is then partitioned as $\boldsymbol{p} = (\boldsymbol{p}_0 , \boldsymbol{p}_1 , \ldots)$.

The vector $\boldsymbol{p}$ is the solution of equations

$$\boldsymbol{pQ} = 0, \quad \boldsymbol{pe} = 1 \tag{1}$$

Where $\boldsymbol{e}$ and $\boldsymbol{0}$ are vectors of all ones and zeros, respectively, and $\boldsymbol{Q}$ is the transition rate matrix of the Markov process which will be obtained for each allocation strategy.

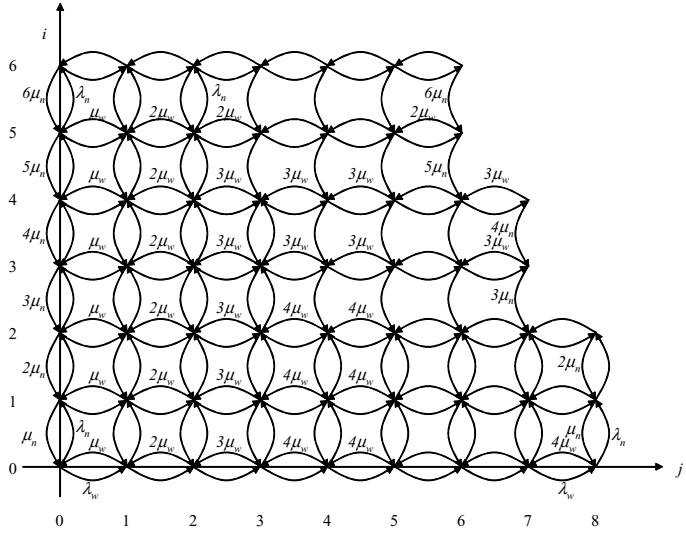


Fig. 2. The state diagram of narrowband and wideband calls occupancy for C=10, G_n=2,G_w=4, B_w=8 and m=2

The state diagram of a system under this system model is shown in Fig. 2. From this figure, we can obtain the transition rate matrix $\boldsymbol{Q}$ of the Markov process

$$Q=\begin{bmatrix} A_0 & D & & & & \\ B_1 & A_1 & D & & & \\ & B_2 & A_2 & D & & \\ & & B_3 & A_3 & D & \\ & & & \bullet & \bullet & \bullet \\ & & & & & \bullet \end{bmatrix} \tag{2}$$

All the solution techniques rely on setting up two-dimensional balance equation for Fig. 2. Let $p_{i,-1}=0$ for $0 \le i \le C-G_w$ and $p_{-1,j}=0$ for $0 \le j \le \lfloor (C-G_n+B_w)/m \rfloor$. We show some balance equations as follows.

$$\begin{aligned} &0 \le i \le G_n-1,\ 0 \le j \le \lfloor (C-G_n-1)/m \rfloor: \\ &[\lambda_n + i\mu_n + \lambda_w + j\mu_w]p_{ij} = \lambda_n p_{i-1,j} + (i+1)\mu_n p_{i+1,j} + \lambda_w p_{i,j-1} + (j+1)\mu_w p_{i,j+1} \\ &0 \le i \le G_n-1, \lfloor (C-G_n)/m \rfloor \le j \le \lfloor (C-G_n+B_w-1)/m \rfloor: \\ &[\lambda_n + i\mu_n + \lambda_w + \lfloor (C-G_n)/m \rfloor \mu_w]p_{ij} = \lambda_n p_{i-1,j} + (i+1)\mu_n p_{i+1,j} + \lambda_w p_{i,j-1} + \lfloor (C-G_n)/m \rfloor \mu_w p_{i,j+1} \\ &0 \le i \le G_n-1, j = \lfloor (C-G_n+B_w)/m \rfloor: \\ &[\lambda_n + i\mu_n + \lfloor (C-G_n)/m \rfloor \mu_w]p_{ij} = \lambda_n p_{i-1,j} + (i+1)\mu_n p_{i+1,j} + \lambda_w p_{i,j-1} \end{aligned} \tag{3}$$

Equations (3) maybe written concisely in matrix form. To do this define a set of $(C-G_w)$-elements row vector $\boldsymbol{p}_i$

$$\boldsymbol{p}_i \equiv [p_{i0}, p_{i1}, p_{i2},\ldots] \tag{4}$$

From above equations (4), we can define submatrices for $i, j = 0,1,\ldots,C-G_w$, $0 \le l \le \lfloor (C-G_n+B_w)/m \rfloor$ by

$$A_l(i,j)=\begin{cases} \lambda_n & if\ i=j-1\ and\ (0 \le i < G_n \mid i < C-l*m) \\ j\mu_n & if\ i=j+1\ and\ i \le C-l*m+B_w \\ a_i(j) & if\ i=j \\ 0 & otherwise \end{cases} \tag{5}$$

$$D(j,k)=\begin{cases} \lambda_w & if\ i=j\ and\ i \le C-l*m+B_w \\ 0 & otherwise \end{cases} \tag{6}$$

$$B_l(i,j)=\begin{cases} \min(l, \lfloor (C-G_n)/m \rfloor, \lfloor (C-i)/m \rfloor)\mu_w & if\ i=j\ and\ i \le C-l*m+B_w \\ 0 & otherwise \end{cases} \tag{7}$$

Where $a_i(j)$ is the value that makes the sum of the row element s of $\boldsymbol{Q}$ equal to zero.

To solve (1) with this transition rate matrix $\boldsymbol{Q}$, we apply the matrix-geometric solution technique based on Neut's solution process. First we find $\boldsymbol{Q}$ matrix by solving the equation

$$\boldsymbol{R} = [\boldsymbol{D}+\boldsymbol{R}^2\boldsymbol{B}_{n1}][\boldsymbol{I}-\boldsymbol{A}_{n1}]^{-1} \tag{8}$$

We now start with a trial solution such as $\boldsymbol{R}=0$ and again iterate until

$$|\boldsymbol{R}(n+1)-\boldsymbol{R}(n)| < \tag{9}$$

Second, find the vector $\boldsymbol{p}_0, \boldsymbol{p}_i$ by solving the equation (10)

$$\boldsymbol{p}_0 = \boldsymbol{p}_0[\boldsymbol{A}_0 + \boldsymbol{R}\boldsymbol{B}_1] \tag{10}$$

$$\boldsymbol{p}_0[\boldsymbol{I}-\boldsymbol{R}]^{-1}\boldsymbol{e}_{\mathrm{m}}^{\mathrm{T}} = 1$$

$$\boldsymbol{p}_i = \boldsymbol{p}_0 \boldsymbol{R}^i$$

Since all $\boldsymbol{p}_i$ can be expressed in terms of $\boldsymbol{p}_0$ by solving the equation (10) recursively, CBP_n and CBP_w can be easily obtained. Let T_n be the admission threshold of narrowband traffic. The new call blocking probability of narrowband traffic P_n^{nb} is given by

$$P_n^{nb} = 1 - \left[\sum_{i=0}^{G_n-1} \sum_{j=0}^{C-G_n+B_w} p_{ij} + \sum_{i=G_n}^{C-G_w-1} \sum_{j=0}^{i+j<T_n} p_{ij} \right] \tag{11}$$

A narrowband hand-off call is accepted if the channels are available. Thus, the hand-off call blocking probability of narrowband traffic P_n^{hb} is given by

$$P_n^{hb} = 1 - \left[\sum_{i=0}^{G_n-1} \sum_{j=0}^{C-G_n+B_w} p_{ij} + \sum_{i=G_n}^{C-G_w-1} \sum_{j=0}^{i+j<C} p_{ij} \right] \tag{12}$$

Let T_w be the admission buffer threshold of wideband traffic. The new call blocking probability of wideband traffic P_w^{nb} is given by

$$P_W^{nb} = \left[\sum_{i=0}^{G_n} \sum_{j=T_w}^{B_w} p_{i,C-G_n+j} + \sum_{i=G_n+1}^{C-G_w} \sum_{j=T_w}^{B_w} p_{i,C-i+j} \right] \tag{13}$$

A wideband hand-off call is accepted if the buffers are available. Thus, the hand-off call blocking probability of wideband traffic P_w^{hb} is given by

$$P_W^{hb} = \left[\sum_{i=0}^{G_n} p_{i,C-G_n+B_w} + \sum_{i=G_n+1}^{C-G_w} p_{i,C-i+B_w} \right] \tag{14}$$

These CBP values are used for computing adaptive guard channels for each class in resource management described in next chapter.

4 Adaptive Resource Management

In our CAC algorithm described in chapter 2, we should compute G_n and G_w. In this section, we describe how to decide reservation partition for narrowband and wideband calls. These reservation partitions play a very important role in admission control and resource utilization, so that it is very critical problem to set these values properly. Each service call reservation partition is allocated with traffic behavior as well as fairness level. Each reservation partition has minimum channel pool to guarantee minimum resource to each service. Using minimum channel pool, it prevents all resources from being occupied by one service so that it can help to make CBP ratio balance and achieve fairness for resource usage.

4.1 Initialization

Initial reservation partition for each class is proportionally allocated with the offered load per cell, which is defined as *call generation rate * required bandwidth units * average call staying time* in a cell. So, the much resource is allocated to class with larger offered load than class with small offered load. The minimum channel pools for each class is set as the same as initial reservation partition. The computations are given by the following equations.

$$G_n = C * \frac{Offered_load_narrow}{\sum offered_load_narrow, offered_load_wide} * \alpha$$

$$G_w = C * \frac{Offered_load_wide}{\sum offered_load_narrow, offered_load_wide} * \alpha$$

where, α has the value between 0 and 1.

4.2 Adjustment

Based on CBP_ratio, resource management algorithm works as follows. If CBP_ratio is greater than threshold, the reservation partition for underprivileged class is incremented up to sum of resource for expecting on-going calls over next term and required resources for CBP fairness. The former is calculated by call's departure rate and the latter is done by call's arrival rate. The required resource for CBP fairness are defined as calls_for_CBP_fairness * BU. The term calls_for_CBP_fairness means the relative number of calls, which should be admitted to make current unfair CBP fairness equal or similar. To compute these calls, at first current available resource is accomplished. This calculation is based on current occupied resource. Once calculating this resource, the remaining resources are partitioned depending on CBP fairness level and offered traffic load. That is, the CBP fairness level has a great important role in deciding call admission during next term. For example, if the CBP unfairness is serious, the bandwidths accommodating all expected calls are to be allocated. On the contrary, if the CBP unfairness is light, bandwidths for admitting several calls among all expected calls are to be demanded. As a result, it is very important to predict the number of calls for getting adequate resources for CBP fairness. Once completing estimation, system calculates how many calls should be admitted in order to lower CBP of unprivileged calls up to current CBP of privileged calls.

The detail procedures for this computation are as follows. In order to make CBP fair, the calls_for_CBP_fairness are calculated by (1 - CBP of other service) * available_resource. This concept is based on biased coin method [10 – 13]. That is, the CBP restricts to the admitted calls over next terms. So, the CBP fairness can be gradually achievable. For better understand, we take an example. We assume that the CBP of service i is 1 / 5 and CBP of service j is 1 / 4. And, the available resources are 20 units where offered traffic load of each class is given as 1 and 2, respectively. According to our algorithm, the acceptance threshold of service i and j in next interval are set to (1 – (1 / 4))* 20 * 1/3 = 5 and (1 – (1 / 5)) * 20 * 2/3 ≅ 11, respec-

tively. So, during next interval, 5 units are reserved for service i and 11 units are reserved for service j. The remaining 4 units are competed with two service class equally. Using this method, the CBP unfairness is gradually to be balanced.

```
If CBP_ratio > threshold  and Then
   available_resource = C - ΣRi
   Gi = Ri + (1 - CBPj)* available_resource * of-
fered_loadi / Σoffered_load
   Gj = Rj + (1 - CBPi)* available_resource * of-
fered_loadj / Σoffered_load
Else
   Nothing is done
```

5 Numerical Analysis

This section presents our numerical analysis for performance of our scheme in the aspects of CBP fairness and resource utilization. The analysis is done with IMOSP. The cell capacity accommodates 20 units. The analysis environments are designed according to offered traffic load as shown in Table 1 with $G_n=2$, $G_w=4$, $B_w=8$ *and* $m=2$.

As you can see in Fig.3-(a) – Fig. 3-(b), both short-term and long-term CBP fairness are only achieved under ours. IMOSP shows slow convergence for long-term CBP fairness. So, it has difficulty to guarantee short-term CBP fairness. These figures also indicate that ours has the lower CBP of wideband calls than IMOSP in long-term period. It is because more narrowband calls are blocked than others so that remaining bandwidth can be more used for wideband calls. The results for case 2 are very noticeable. As you can see Fig. 4-(a) – Fig. 4-(b), IMOSP and ours has a big difference in CBP fairness. In Case 2, the traffic intensity between wideband and narrowband is wide. Especially, wideband call arrives with large traffic. IMOSP shows an obvious CBP unfairness between wideband call and narrowband call. On the other hand, our scheme shows a fair CBP between two services. We can observe from Fig. 4-(b) that the CBP of wideband decreases as the CBP of narrowband increases. After all, two CBP are converged into their average CBP value. Fig. 5-(a) – Fig. 5-(b) show resource occupied by narrowband and wideband vs. link capacity. This factor can be considered as resource fairness partially. In IMOSP, narrowband calls occupy 10% of total capacity in latter part. However, ours shows more fair resource usage than IMOSP. It is mostly because we make use of minimum channel pool concept, which cannot be occupied by other services. It prevents all resources from being occupied by one service class.

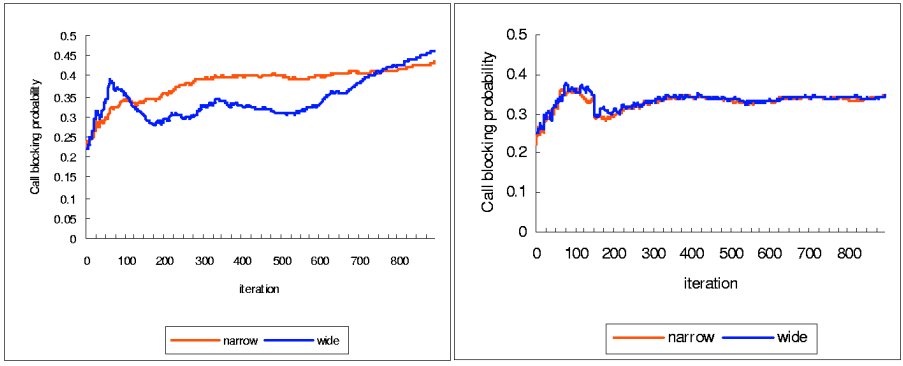


Fig. 3. (a) IMOSP in Case 1, (b) Ours in Case 1

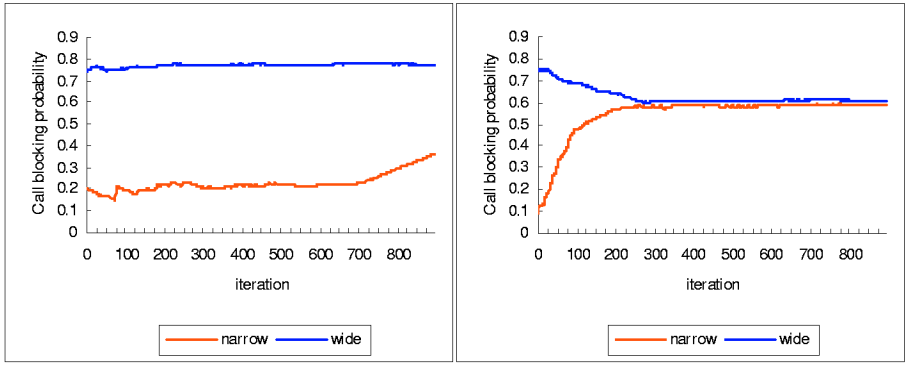


Fig. 4. (a) IMOSP in Case 2, (b) Ours in Case 2

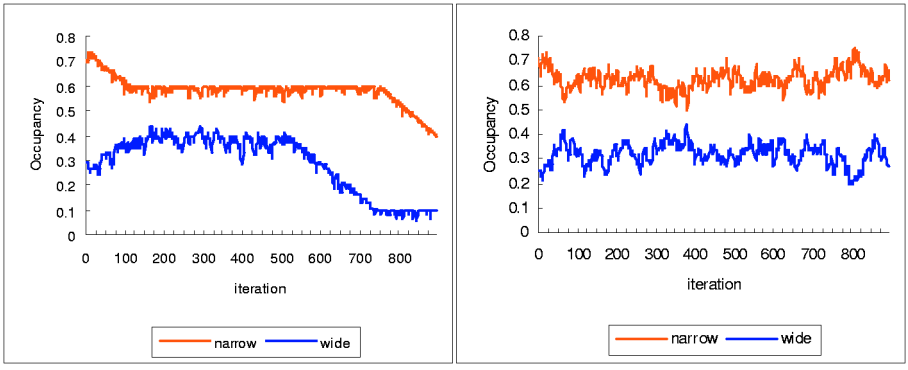


Fig. 5. (a) Resource occupancy in IMOSP, (b) Resource occupancy in Ours

Table 1. Traffic values for analysis

Type of call	Parameters	Case 1	Case 2
narrowband	Call arrival rate (call/sec)	0.5	0.2
	Required bandwidth (unit)	1	1
	Service time in a cell (sec)	4	2
wideband	Call arrival rate (call/sec)	0.1	1
	Required bandwidth (unit)	2	2
	Service time in a cell (sec)	4	4

6 Conclusion

This paper proposed a novel CAC scheme and resource management algorithm that guarantee both short-term and long-term fairness between heterogeneous services with different traffic properties and enhance resource utilization of system. The proposed method has been analyzed using a two-dimensional Markov chain and Neut's matrix-geometric solutions. By numerical analysis, we demonstrated that our CAC scheme actually achieves fair admitting probability for wideband and narrowband calls and also improves resource utilization regardless of traffic behavior.

References

1. M. Naghshineh et al., "Distribued call admission control in mobile/wireless networks," IEEE JSAC, Vol. 15, May 1996, pp. 1208 – 1225.
2. X.Y. Luo et al., "A dynamic measurement-based bandwidth allocation scheme with QoS guarantee for mobile wireless networks," IEEE WCNC'00, September 2000.
3. S. Choi et al., "Predictive and adaptive bandwidth reservation for handoffs in QoS-sensitive cellular networks," ACM SIGCOMM'98, 1998, pp. 254 – 275.
4. J. Misic et al., "Admission control for wireless networks with heterogeneous traffic using event based resource estimation," IEEE ICCCN'97, September 1997.
5. F. Prihandoko et al., "Adaptive call admission control for QoS provisioning in multimedia wireless networks," Journal of Computer Communications, Elsevier Publisher, November 2002.
6. Y. Xiao et al., "Optimal Admission Control for Multi-Class of Wireless Adaptive Multimedia Services", IEICE Transactions on Communications, Special Issue on Mobile Multimedia communications, Vol. E84-B, No.4, April 2001, pp.795-804.
7. M. Naghshineh and A.S. Acampora, "QoS Provisioning in Micro-Cellular Networks Supporting Multiple Classes of Traffic," Wireless Networks, vol. 2, pp. 195-203, 1996.
8. J. Y. Lee et al.,"Realistic Cell-Oriented Adaptive Admission Control for QoS Support in Wireless Multimedia Networks," IEEE Trans. Vehicular Technology, Vol. 52, No. 3, May 2003.
9. B. M. Epstein et al., "Predictive QoS-based admission control for multiclass traffic in cellular wireless networks," IEEE JSAC, Vol. 18, No. 3, March 2000, pp. 523 – 534.
10. L. J. Wei, "The Adaptive Biased Coin Design for Sequential Experiments," Journal of Annals of Statistics, Vol. 6, Jan. 1978, pp. 92 – 100.

11. J. M. Steele, "Efron's Conjecture on Vulnerability to Bias in A Method for Balancing Sequential Trials," Biometrika, 67, pp. 503 – 504.
12. S.J. Pocock, Clinical Trials : A Practical Approach, John Wiley & Sons Ltd., 1991, pp. 79–80.
13. B. Efron, "Forcing a Sequential Experiment to be balanced," Biometrika, 58, pp. 403 – 417.
14. M. Schwartz, *Broadband Integrated Networks*, Prentice Hall, 1996.
15. M.F. Neuts, *Matrix-Geometric Solutions in Stochastic Models*, Johns Hopkins University Press, 1981.

Application of MCDF Operations in Digital Terrain Model Processing

Zhiqiang Ma[1,2], Anthony Watson[3], and Wanwu Guo[3]

[1] Department of Computer Science, Northeast Normal University
138 Renmin Street, Changchun, Jilin, China
mazq@nenu.edu.cn
[2] School of Computer Science, Jilin University
Changchun, Jilin, China
[3] School of Computer and Information Science, Edith Cowan University
2 Bradford Street, Mount Lawley, Western Australia 6050, Australia
{a.watson, w.guo}@ecu.edu.au

Abstract. Modified conjugate directional filtering (MCDF) is a new method proposed by Guo and Watson in 2002 for digital data and image processing. It provides ability in not only integrating directional-filtered results in conjugate directions into one image that shows the maximum linear features in these conjugate directions, but also further manipulating the outcomes using a number of predefined MCDF operations for different purposes. Digital terrain model (DTM) has brought new dimensions to the use of geographic data. Since MCDF operations are based on directional filtering, naturally these operations should reveal the local changes in elevation when they are applied to DTM data. MCDF operations can also keep both textual and enhanced structural information in the same image, and have ability to produce pseudo 3D views, so theoretically this new method should be applicable to and useful in DTM data processing. In this paper, we discuss the results of using other existing methods to DTM processing. The results of applying MCDF(add1) and MCDF(add3) to the same DTM data are then presented for making comparisons with the results from other means.

1 Introduction

Modified conjugate directional filtering (MCDF) is a new method proposed by Guo and Watson [1] for digital data and image processing. By using MCDF, directional-filtered results in conjugate directions can be not only merged into one image that shows the maximum linear features in the two conjugate directions, but also further manipulated by a number of predefined MCDF operations for different purposes. Tests of using MCDF for processing aerial photographs [2], airborne magnetic data [3], and X-ray radiograph [4] have shown that it provides a new and useful means for digital data and image processing. MCDF not only combines the enhanced features in two conjugate directions together with further manipulation though adjustable weighting factors, but also retains the background information on the original image. This cannot be achieved by using any single conventional method for linear enhancement [5][6][7].

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 471–478, 2004.
© Springer-Verlag Berlin Heidelberg 2004

Traditionally, geographic elevations are compiled as topographic maps in different regions as one of the fundamental data for land management, city planning, military training, geosciences surveys, and other fields. By counting the number of contours in a topographic map, geographic variations in an area can be estimated. A geographic location can also be found using its specific contour pattern and in reference to the features in the surrounding areas. Digital terrain model (DTM), or digital elevation model, has brought new dimensions to the use of geographic data. By using DTM data, geographic variations in an area can be presented as 3D images that give intuitive views of the natural variations in the area. Topographic map of the area can also be plotted easily as 2D contour map using the same DTM data. Since MCDF operations are based on directional filtering, naturally these operations should reveal the local changes in elevation when they are applied to DTM data. MCDF operations can also keep both textual and enhanced structural information in the same image, and have ability to produce pseudo 3D views, so theoretically this new method should be applicable to and useful in DTM data processing. In this paper, we first briefly present the concepts of the MCDF operations, and then discuss the results of using other existing methods to DTM processing. The results of applying MCDF(add1) and MCDF(add3) to the same DTM data are then presented for making comparisons with the results from other means.

2 MCDF Operations

Directional filtering is used to enhance linear features in a specific direction [5][6][7]. In some cases, identifying conjugate linear information on an image is particularly concerned. Directional filtering can be made in two specific conjugate directions to enhance these conjugate features. Normally the filtered results from the two conjugate directions are shown on two separate images. This is inconvenient for revealing the relationships between linear features in the two conjugate directions. The linear enhancement using directional filtering is achieved by constraining or removing the textural features or low-frequency components from the original image to outline the structural features or high-frequency components contained in the original image. Thus, directionally filtered image often lacks contrast depth because most background information is removed.

These two weaknesses of using the conventional directional filtering are overcome by MDCF method, which firstly combines two (or more) directional-filtered results in conjugate directions into one image that exhibits the maximum linear features in the two conjugate directions, and secondly retains the background information by superimposing the directionally filtered data onto the original data. Therefore, the analytical tests should be designed in a way through which these two improvements can be clearly revealed.

Assuming f_0 to be the original data file, f_1 and f_2 to be the directional-filtered data files in the two conjugate directions, the general operation of the MCDF can be expressed as [1]

$$\text{MCDF} = W_0 \cdot f_0 + F_2[W_1 \cdot F_1(f_1), W_2 \cdot F_1(f_2)]; \tag{1}$$

where W_0, W_1 and W_2 are selective constants; F_0, F_1 and F_2 are pre-defined generic functions. Consequently, some MCDF operations are defined using formula (1) as

$$\text{MCDF(add1)} = W_0 \cdot f_0 + W_1 \cdot f_1 + W_2 \cdot f_2; \tag{2}$$

$$\text{MCDF(add2)} = W_0 \cdot f_0 + \text{abs}(W_1 \cdot f_1 + W_2 \cdot f_2); \tag{3}$$

$$\text{MCDF(add3)} = W_0 \cdot f_0 + W_1 \cdot \text{abs}(f_1) + W_2 \cdot \text{abs}(f_2); \tag{4}$$

$$\text{MCDF(max1)} = F_0(W_0 \cdot f_0) + \max(W_1 \cdot f_1, W_2 \cdot f_2); \tag{5}$$

$$\text{MCDF(max2)} = F_0(W_0 \cdot f_0) + \max[W_1 \cdot \text{abs}(f_1), W_2 \cdot \text{abs}(f_2)]; \tag{6}$$

$$\text{MCDF(ampl)} = W_0 \cdot f_0 + \text{sqrt}(W_1 \cdot f_1 \cdot f_1 + W_2 \cdot f_2 \cdot f_2). \tag{7}$$

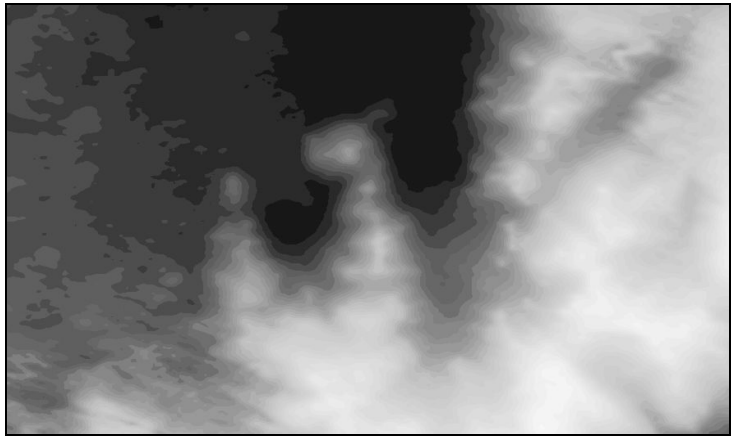
Some analytical results have verified that the MCDF operations not only enhance the conjugated features in both conjugated directions in an image, but also retain the low-frequency information in the original image [8]. Table 1 shows the statistical results of spectral analysis over a digital terrain model using MCDF(add1) [8]. It is evident that the MCDF(add1) operation has enhanced the highest-frequency component by 9 times from its relative intensity of 0.5% in the original image to 4.5% in the MCDF(add1) image. This is achieved by keeping almost no change in the maximum intensity and standard deviation in both images, which means that there is almost no loss in low-frequency components in the MCDF(add1) image. The medium-frequency components are also intensified from 6.3% in the original image to 16.9% in the MCDF(add1) image, an increase of 2.7 times. By keeping the same low-frequency components, bringing a moderate increase in medium-frequency components, and elevating high-frequency components by at least 9 times, all together the MCDF(add1) operation makes not only features in the NE and NW directions in the MCDF(add1) image look more prominent, but also the whole image appear richer in contrast depth and thus more smooth.

Table 1. Statistics of radial spectra of the original DTM and its MCDF(add1) images

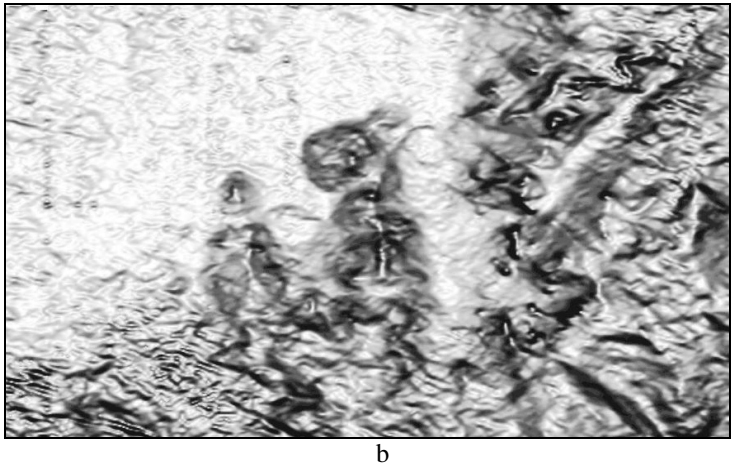
Statistics	Original image		MCDF(add1) image	
	Absolute	Relative (x/Max)	Absolute intensity	Relative (x/Range)
Min (high-frequency components)	826	0.5%	7446	4.5%
Max (low-frequency components)	164359	100%	164345	100%
Median (medium-frequency components)	10372	6.3%	27810	16.9%
Std	26299	16%	25492	15.5%

3 Processing DTM Using Conventional Operations

Figure 1a is the grayscale DTM image in central Australia. This region has a relatively low topographic relief (<200 m). The dark colors indicate the desert whereas the light colors indicate the highlands or hills in the desert. With the dominance of dark colors in the desert, detailed features within the desert are hardly seen on the original image. Figure 1b is the grayscale 3D relief image of the same model. The illumination is applied vertically so that no shading effect bias to any direction is made. This image gives a strong impression on the topographic relief of the region that cannot be seen in the original DTM image. However, we should notice that whereas some subtle variations in the desert are exposed on this 3D image, it is hardly to estimate the real scale of the terrain relief on showing, just as we cannot estimate how high a tall building is if we look down right over it in the air.



a



b

Fig. 1. Original DTM image (a) and the 3D image illuminated vertically (b)

More interesting observations come from the changes in illuminating orientation. Figure 2a shows a 3D shaded-relief image of the same model as being illuminated from northeast at an angle of 45°, whereas Figure 2b is the result as being illuminated from northwest at an angle of 45°. For the same model, being illuminated from different directions, it is expected that images with subtle differences are produced, but what is unexpected is that the differences between these two images are so significant that one could easily regard them as showing two different regions. When comparing these two images with Figure 1b, one would question which 3D image should be regarded as the 'true' presentation of the DTM.

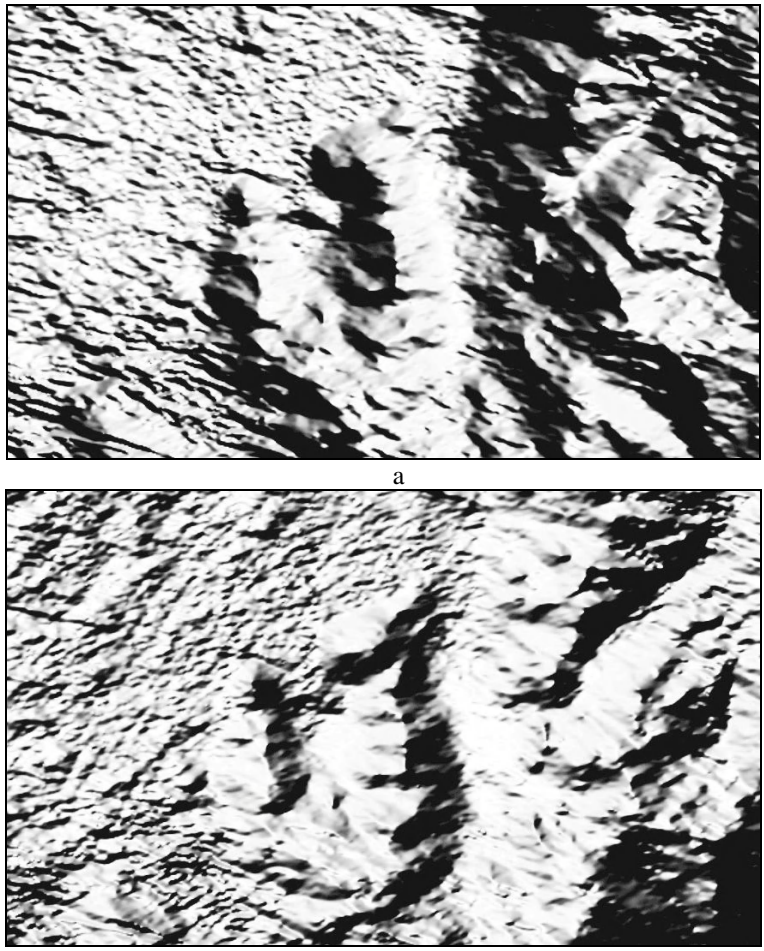


Fig. 2. Shaded-relief DTM images illuminated from northeast (a) and northwest (b)

Figure 3 shows the image after applying a conventional edge detection operation on the same terrain model. It looks like a real topographical map with contours indicating the terrain relief. This image gives us some thoughts on the real scale of the terrain relief in this region by counting the number of contours. However, the color pat-

tern as direct indication of terrain relief shown in the original image (Fig. 1a) is lost; furthermore, the impressive 3D shaded-relief features cannot be retained to some extent. This implies that by using any single conventional operation we can have an enhanced image that shows either quantitative information as contours, or qualitative information as color patterns or 3D shades on terrain relief, but cannot contain both even to some extent.

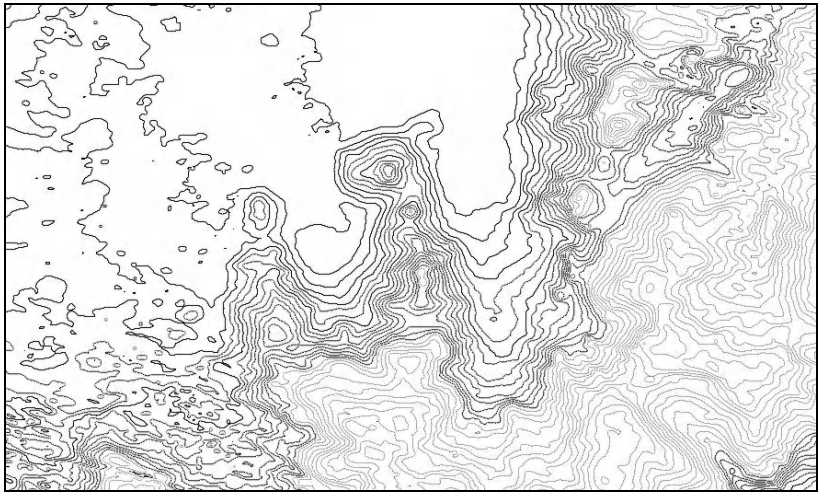


Fig. 3. Edge detected DTM image of the original terrain model

4 Processing DTM Using MCDF

Although several MCDF operations are defined [1], we only use MCDF(add1) and MCDF(add3) to process the DTM data. The MCDF(add1) operation can produce an image with 3D effect to some extent, which is especially useful in presentation of terrain relief. The MCDF(add3) operation is able to outline the gradient changes that indicate the topographic variation in DTM.

Firstly, directional filtering is applied in both the NE and NW directions to the original DTM data. The output (f_{NE} and f_{NW}) along with original data is then used to produce the MCDF images. Figure 4 shows the image processed using MCDF(add3) with parameters of $W_0 =$ and $W_1 = W_2 = 3$ (Exp. (3)). Essentially, the MCDF(add3) image is equivalent to a combination of the original DTM image (Fig. 1a) and the edge-detected DTM image (Fig. 3). As a result, both color patterns and contours are contained in this image. As we mentioned before, any single traditional operation is not capable of producing such an image. Since the directional filtering is applied in the conjugated NE and NW directions, linear features in these two directions are especially enhanced. However, this image does not show 3D effect to any extent.

Figure 5 shows the image processed using MCDF(add1) (Exp. (2)). The parameters for this operation are: $W_0 = 1$ and $W_1 = W_2 = 3$. In addition to the enhanced features shown in the MCDF(add3) image (Fig. 4), the MCDF(add1) image shows 3D effect to some extent, although its impression is far less than any 3D shaded-relief im-

age presented previously in this paper. However, we should notice that the MCDF operations indeed indicate the capability in integrating information in different patterns and formats into a single image, of which any single traditional operation is not capable.

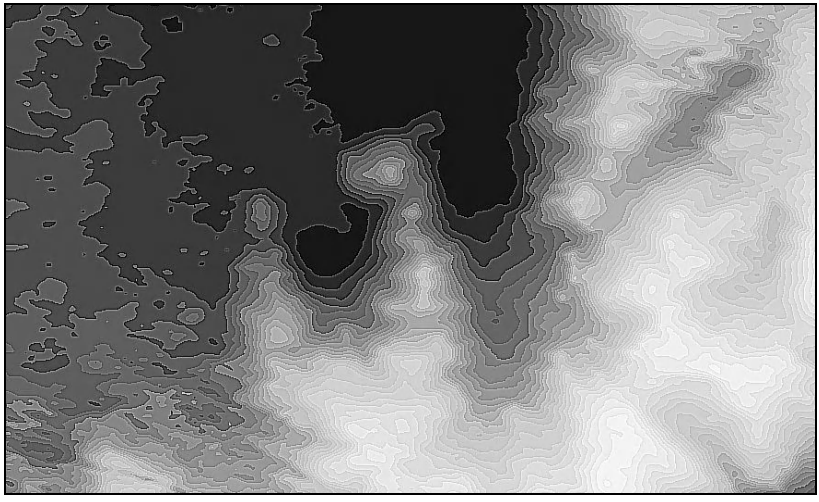


Fig. 4. DTM image processed using MCDF(add3)

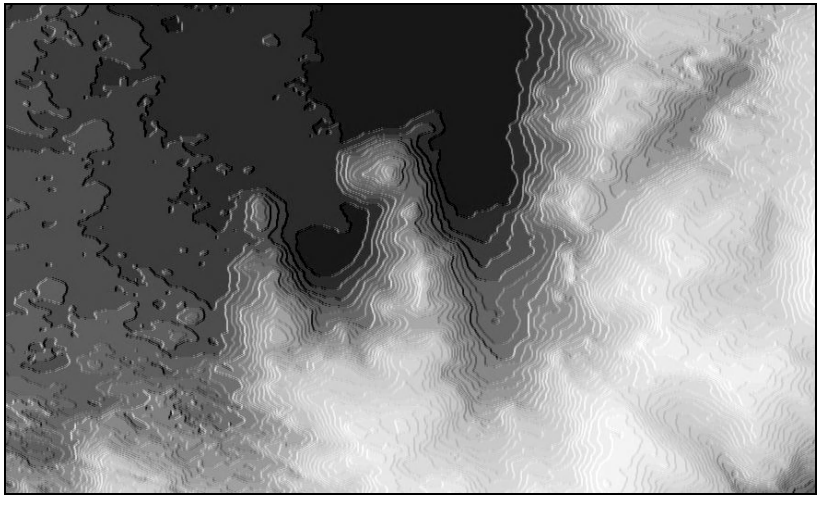


Fig. 5. DTM image processed using MCDF(add1)

5 Discussion and Conclusion

Both conventional and MCDF operations have been used to process a DTM image. The 3D shaded-relief method can produce visually impressive images, but the differences in presentation caused by applying illumination from different directions on the

same model are so great that these images can be easily regarded as representing different models (Figs. 1b & 2). The 3D shaded-relief images are better used to only present qualitative information on terrain relief because the virtual relief on such a 3D image is either exaggerated or diminished in different areas under the same illumination scheme, or in the same area under different illumination schemes.

Edge detection operation can produce a relatively quantitative presentation of terrain relief by means of contours. However, no color pattern and 3D effect can be retained on such an edge-detected image (Fig. 3). Any single conventional operation is hardly able to produce an image that contains both qualitative and quantitative forms of information.

MCDF operations are able to partly combine both qualitative and quantitative forms of information into a single image. The MCDF(add3) image contains both contours and colors (Fig. 4), whereas the MCDF(add1) image shows terrain relief information in forms of contours, colors, and 3D effect (Fig. 5). Therefore, MCDF adds new dimensions to DTM processing.

Acknowledgements. We are grateful to the Northern Territory Geological Survey of Department of Mines and Energy of Australia for providing us the DTM data. The Faculty of Communication, Health and Science of the Edith Cowan University is thanked for supporting this research project. Dr W Guo's visit to NENU computer science department was supported by The Northeast Normal University. The constructive comments made by the anonymous referees are acknowledged.

References

1. Guo, W., Watson, A.: Modification of Conjugate Directional Filtering: from CDF to MCDF. Proceedings of IASTED Conference on Signal Processing, Pattern Recognition, and Applications. Crete, Greece (2002) 331-334.
2. Watson. A., Guo, W.: Application of Modified Conjugated Directional Filtering in Image Processing. Proceedings of IASTED Conference on Signal Processing, Pattern Recognition, and Applications. Crete, Greece (2002) 335-338.
3. Guo, W., Watson, A.: Conjugated Linear Feature Enhancement by Conjugate Directional Filtering. Proceedings of IASTED Conference on Visualization, Imaging and Image Processing. Marbella, Spain (2001) 583-586.
4. Guo, W., Watson, A.: Medical Image Processing using Modified Conjugate Directional Filtering. The IASTED International Conference on Biomedical Engineering, Salzburg, Austria (2003).
5. Jahne, B.: Digital Image Processing: Concepts, Algorithms and Scientific Applications. Springer-Verlag, Berlin Heidelberg (1997).
6. Proakis, J.G., Manolakis, D.G.: Digital Signal Processing: Principles, Algorithms and Applications. Prentice-Hall, Upper Saddle River New York (1996).
7. Richards, J.A.: Remote Sensing Digital Image Analysis. Springer-Verlag, Berlin Heidelberg (1993).
8. Kong, J., Zhang, B., Guo, W.: Analytical test on effectiveness of MCDF operations. Computational Science, Lecture Notes in Computer Science, Springer-Verlag, Berlin Heidelberg (2004).

Visual Mining of Market Basket Association Rules

Kesaraporn Techapichetvanich and Amitava Datta

School of Computer Science & Software Engineering
University of Western Australia
Perth, WA 6009
Australia
{kes,datta}@csse.uwa.edu.au

Abstract. Data mining is increasingly becoming important in extracting interesting information from large databases. Many industries are using data mining tools for analyzing their vast databases and making business decisions. Mining association rules is an important data mining method where interesting associations or correlations are inferred from large databases. Though there are many algorithms for mining association rules, these algorithms have some shortcomings. Most of these algorithms usually find a large number of association rules and many of these rules are not interesting in practice. Hence, there is a need for human intervention in mining interesting association rules. Moreover, such intervention is most effective if the human analyst has a robust visualization tool for mining and visualizing association rules. In this paper we present a three-step visualization method for mining market basket association rules. These steps include discovering frequent itemsets, mining association rules and finally visualizing the mined association rules. Most previous visualization methods have concentrated only on visualizing association rules that have been already mined by using existing algorithms. Our method allows an analyst complete control in mining meaningful association rules through visualization of the mining process.

Keywords: Information Visualization, Association rule, Market Basket, Data Mining

1 Introduction

Mining association rules is a well researched area within data mining [5]. There are many algorithms for generating *frequent itemsets* and *mining association rules* [1,13,11]. Such algorithms can mine association rules which have *confidence* and *support* higher than a user-supplied level. However, one of the drawbacks of these algorithms is that they mine all rules exhaustively and many of these rules are not interesting in a practical sense. Too many association rules are difficult to analyze and it is often difficult for an analyst to extract a meaningful (small) set of association rules. Hence there is a need for human intervention during the mining of association rules [1,16] so that an analyst can directly influence the mining process and extracts only a small set of interesting association rules.

Information visualization has been a growing area of research in recent years [3]. It is much easier to visualize and analyze a large database compared to reading it in

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 479–488, 2004.
© Springer-Verlag Berlin Heidelberg 2004

textual form. There are many methods for visualizing higher dimensional databases on standard graphics displays [12,14,9]. Recently, several researchers have investigated the visualization of association rules [17,10,6]. The main motivation behind this research is to identify interesting association rules from a large number of rules mined by using an existing algorithm. Hence, visualization of association rules does not directly address the issue of meaningful intervention so that an algorithm mines only interesting association rules.

We feel that it is important for an analyst to participate in the mining process in order to identify meaningful association rules from a large database. Any such participation should be easy from an analyst's point of view. Hence, visualizing association rule mining seems to be a natural way of directing the mining process. We propose a visualization method that helps an analyst to mine association rules in three steps, *identifying frequent itemsets*, *mining association rules* and *visualizing* the mined association rules. In addition, the analyst has complete control on deciding on the *antecedents* and *consequents* of each rule and the whole process is intuitively simple for an analyst. Though a complete visual mining process is slow compared to an automated process, it has the advantage of exploring only interesting association rules. As we have mentioned before, an automated process can mine many association rules that are not meaningful practically. Our visualization tool is extremely simple to use and avoid screen clutter. This makes it an attractive option to use both for small and large databases. Our interfaces and applications presented in this paper have been implemented by using MFC Visual C++.

The rest of this paper is organized as follows. In section 2, a short review of association rule mining and related research is discussed. We present our technique for mining association rules in section 3. Finally, we conclude with some comments in section 4.

2 Previous Research

2.1 Association Rules

An association rule [1] is a rule of the type $A \Rightarrow B$, where A is an itemset called antecedent, body, or left-hand side (LHS) and B is an itemset called consequent, head, or right-hand side (RHS). The implication of the rule is that if A appears in a transaction, there is a high probability that B will also appear in the same transaction. Each itemset consists of items in a transactional database. Items existing in the antecedent are not in the consequent. In other words, an association rule is of the form $A \Rightarrow B$, where A,B $\subset$ I and A $\cap$ B $= \phi$. I $= \{ i_1, i_2, ..., i_n \}$ is a set of items in the transaction database where i_j, 1 $\leq$ j $\leq$ n, is an item in the database that may appear in a transaction. The two common measures of interestingness are *support* and *confidence*. The rule $A \Rightarrow B$ holds in the transactional database with support s if s is the percentage of transactions that contain $A \cup B$ (both A and B). This is also the probability $P(A \cup B)$. The same rule $A \Rightarrow B$ has confidence c if c is the percentage of transactions containing A that also contain B, i.e., the conditional probability $P(B|A)$. An interesting association rule should have both support and confidence above a user specified level. As an example, consider the association rule {cheese,bread}$\Rightarrow$ {milk,egg} derived from a supermarket transaction database. If this rule has a support of 12%, it means that the four items

{cheese,bread,milk,egg} appear together in 12% of all transactions. If this rule has a confidence of 52%, it means that 52% of all customers who purchased cheese and bread, also purchased milk and egg in the same transaction.

A term, frequent itemset [5], is used to define an itemset whose numbers of co-appearing items in database is greater than a user specified support. In a database of supermarket transactions, a frequent itemset is a set of items frequently purchased together. For example, if the user specified support is 30% and an itemset {cheese, bread, egg} appears in 41% of all transactions, then {cheese, bread, egg} is a frequent itemset.

2.2 Related Work

Information visualization has developed into a major area of research to enhance insight into large databases. From existing work, visualization techniques can be categorized into five groups. First, geometric techniques such as Scatterplot Matrix [2] and Parallel Coordinates [8] try to depict multi-dimensional data on the plane. Second, iconographic techniques use the features of icons or glyphs to represent data variables. The third group is known as hierarchical techniques such as Worlds within Worlds [4]. The fourth group is Pixel-Oriented techniques such as VisDB [9]. The last group is Table-based techniques such as Table Lens [14].

Various visual presentation methodologies of association rules have been developed to represent the results of rules generated from data mining algorithms. Prior research of presenting association rules including visualization approaches can be categorized into three main groups: table-based, matrix-based, and graph-based.

First, in table-based techniques, the columns of a rule table represent the items, the numbers of antecedent and consequent, the support, and the confidence of the association rules. Each row represents the association rule. Some examples of the table-based techniques are included in SAS Enterprise Miner [7] and DBMiner [5]. Second, in matrix-based techniques a two-dimensional matrix or grid represents the antecedents and consequents. The height and colour of columns are used to represent the properties of the association rule such as support and confidence. For example, MineSet [12] uses the 2-D matrix technique to visualize the results of the association rules. Wong et.al [17] use a similar technique to 2-D matrix in which both of the antecedents and consequents are represented by a matrix square based on the x-y coordinates. In their technique, blue and red columns illustrate the antecedent and consequent, respectively. The columns of the confidence and support of the association rules are scaled and plotted at the farthest end of the x-y plane.

The techniques in the last group are based on directed graphs. These techniques use nodes to represent the items and edges to represent the associations of the items in the rules. For example, a rule A $\rightarrow$ B is represented by a directed graph in which A and B are the nodes. The edge connecting A and B has the arrow pointing towards the consequents of the rule. DBMiner [5] applies this technique into its system, called *Ball graph*. The nodes in Ball graph are called balls whose size vary depending on number of items represented.

Some prior work integrate the above techniques into their systems, e.g., CrystalClear [10] is an integrated technique of grid and tree-based representation for visualizing the number of items and the list of antecedents and consequents. Another technique, not being specified in any of the above groups, is *interactive mosaic plots* for visualizing association rules [6]. As its name suggests, this technique applies mosaic visualization technique to represent the relationships among items in each association rule instead of visualizing the results of association rule mining.

None of the prior techniques incorporates visualization to generate frequent itemsets and association rules. All these techniques are used for visualizing the results of the association rules derived from data mining algorithms.

3 Visual Mining of Association Rules

We use a technique called *hierarchical dynamic dimensional visualization* (HDDV) [15] for visualizing each step of the data mining process. We briefly discuss the relevant parts of the HDDV technique which will help us in describing our visual data mining approach. The HDDV technique is useful for visualizing each dimension or attribute of a higher dimensional data set separately and exploring the dependencies among the attributes. This is in contrast to other visualization techniques like parallel coordinates and scatter plots where data points are plotted against two or more dimensions.

In the HDDV technique, each dimension of the dataset is plotted as a horizontal bar, called a *bar stick*, similar to a bar in a histogram. Each bar stick is of equal length to maximize screen space. Suppose a bar stick represents an attribute $attr1$, e.g., age. The bar stick for $attr1$ represents the complete range of values that $attr1$ can take. For example, if $attr1$ is age, the possible range of values may be 0-100. The analysis in HDDV is driven by query range. If there is a query to visualize all records between the ages 30-45, a part of the bar stick is rendered to represent this. The length of the rendered part is proportional to the number of records satisfying the query range. Each bar stick is composed of a sequence of vertical lines and each vertical line is used for representing one or more records. It $total$ is the total number of records in the database and $length$ is the length of a bar stick, then each vertical line within a bar stick represents $total/length$ records. Hence, the HDDV system can be used for visualizing databases of any size by varying the number of records represented by each vertical line within a bar stick.

The system can be used for analyzing a database through hierarchical querying. The user can choose a set of attributes in a hierarchical fashion by specifying query ranges. Suppose $attr1$ and $attr2$ are two attributes chosen by the user in a hierarchical fashion. When the user specifies a range for $attr1$, a part of the bar stick representing the records within this range are rendered in color. Next, the user chooses a range for $attr2$ and a part of the bar stick for $attr2$ is rendered so that all the records in this bar stick satisfy both the ranges for $attr1$ and $attr2$. The hierarchical analysis proceeds in this fashion. We omit the other details of the HDDV system since those are not directly related to the technique discussed in this paper.

3.1 An Overview of Our Technique

Our visual mining technique has three stages. In the first stage, the user can find a suitable *frequent itemset*. In most data mining algorithms, the selection of a frequent itemset is done automatically. Any item that has an occurrence above the user specified support is chosen as the member of the frequent itemset. Though this method is efficient for identifying all the frequently occurring items, the subsequent association rule mining quite often discovers a large number of association rules involving these frequently occurring items. In our technique, we give the user complete control for choosing the frequent item set so that the user can concentrate on *interesting* items.

In the second stage, the user can mine interesting association rules by specifying the antecedents and consequents of each rule from the frequent itemset chosen in the first stage. The user can experiment with different combinations of antecedents and consequents and save a rule if it is interesting. Finally, in the third stage, the user can visualize all the rules saved during the second stage. Our technique helps filtering the uninteresting frequent itemsets and uninteresting association rules by employing human knowledge.

In our application, we split an application window into two areas: left and right panels. The left panel is a user control panel which allows the user to input parameters. The right panel is a visualization panel which renders results depending on the parameters set in the left panel.

3.2 Identifying a Frequent Itemset

This part of our system assists the user to search for a frequent itemset based on a user-specified minimum support. The user can provide the minimum support to filter only items that they are interested in. After specifying the minimum support, all items exceeding the threshold are loaded and sorted in descending order of their support. The user can use the sorted list as a guide in selecting each item in the frequent itemset. Each selected item is represented by a barstick with the percentage of its support. After the first selection of an item, the system generates a list of items that co-exist with the first selected item. All the items in this co-existing item list have supports greater than the user-specified minimum support. The co-existing item list is also generated each time a subsequent item is chosen. The percentage of support is calculated by comparing the numbers of the first and second selected items appearing together with the total number of appearance of the first selected item. At each step, the bar sticks are displayed using the HDDV technique discussed before.

This technique helps users to find items which tend to appear together in the transactions. In addition, the system has user interaction to support the detail of each selected item. On right clicking in each bar, the percentage of each item in the co-existing item list and its support are displayed to help the user make decisions and compare selected interesting items and their supports. As shown in Figure 1, the co-existing item list for cereal consists of 40% of biscuit, 28% of chocolate, and 36% of juice.

The user can change a previously chosen item at any stage of choosing the frequent itemset. Each item in the set is chosen from a drop-down list of items and the user can resize the frequent itemset by deleting the last item at any stage. The user can change any

previously chosen item by successively reselecting any item from any drop-down list. Once the user has finalized the frequent itemset, it can be saved for the later stages of the mining process. We have shown only seven items in Figure 1, however, it is possible to include any number of items in the left panel through a scrolling window.

3.3 Selecting Interesting Association Rules

In this stage, the selected frequent itemset from the first stage is used to generate the association rules. Again, we provide complete freedom to the user for choosing the association rules including the items in the antecedent and consequent of each rule. The number of items in the antecedent and consequent of an association rule is not limited only to one-to-one relationships. The system supports many-to-many relationship rules as well. In Figure 2, milk, bread, cheese, and cereal are the selected frequent itemset. The user is allowed to generate a many-to-many relationship rule namely milk and bread as antecedent and cheese and cereal as consequent. The first colored bar illustrates the proportion of selected items, milk and bread for an antecedent. The second colored bar represents all selected items of an association rules or in other words it shows the proportion of the consequent items, cheese and cereal, appearing together with the antecedent of the rule. In the left control panel, the system shows the support of antecedent, the support of the selected itemset, and the confidence of the association rule. The user can save a rule when the antecedents and consequents are finalized.

3.4 Visualizing Association Rules

This part deals with visualization of the mined association rules in the second stage. The visualization allows analysts to view and compare the mined association rules generated from the first two steps. Among the selected interesting rules, the visualization bars allow analysts to obtain the most significant and interesting rules. Figure 3 represents three association rules. For example, the first rule shows the relationship of the antecedent: milk and bread and the consequent: cheese and cereal.

The confidence, the antecedent support, and the itemset support of this rule is 49, 51, and 25, respectively. For the second rule, the first bar, support 51, represents the antecedent: milk and bread and the second bar, support 40, represents cheese. The confidence is 78. The antecedent support of the last rule is 30, the frequent of itemset is 25, and the confidence is 83. From the result of visualizing, the last rule has the highest confidence while the antecedent support is the lowest.

3.5 Data Structures Used in Our System

We used a synthetic supermarket (market basket) transaction database for our experiments. Our algorithm scans a market basket transaction database twice. The first scan is to count the support of each item in the transaction records. The second scan is to generate a bitwise table to store the item lists of the original transaction records. We use a bitwise operation in representing both existing and non-existing items. In the first stage for identifying the frequent itemset, an item identification list including non-existing items of each transaction is converted to a bit-vector representation, where 1 represents

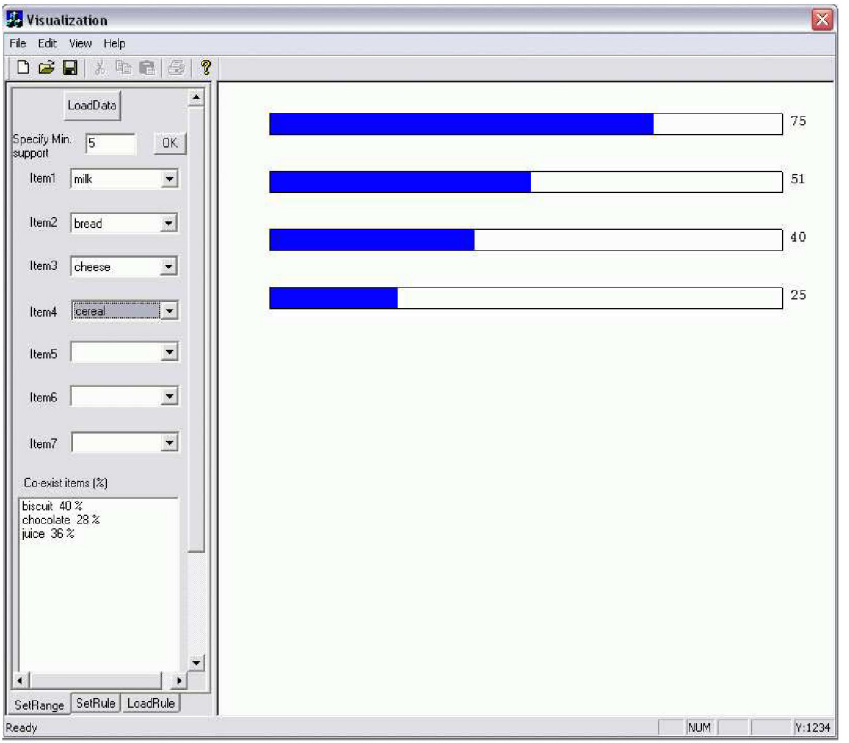


Fig. 1. The right drawing space represents each selected item as a barstick with its support. The control tab represents combobox for each selected item and the list of its co-existing items.

an existing item and 0 represents a non-existing item in the record. For example, suppose a market basket transaction database contains four items milk, bread, cheese, and cereal in ascending order of item identifications and a transaction contains two items: milk and cheese. A bit-vector of this transaction is 1010. Hence, the associated items can be retrieved by applying a bitmask operation to each transformed item list. Each bitmask is generated by transforming all selected items to bits which are set to 1. After selecting each interesting item from a menu list in the first stage, an associated item list is generated to support the user's search for the next interesting item. To reduce searching time of associated items in each transaction, the associated item list contains only the indexes of transactions with all selected items appearing together. Each transaction index is linked to the bitwise table so that all associated items in that transaction can be retrieved.

This technique can support a large number of items in a transaction database. Though, the bitwise technique needs some preprocessing time to convert the transaction records to a bitwise table, it is more efficient and effective to search the existing and associated items at run time.

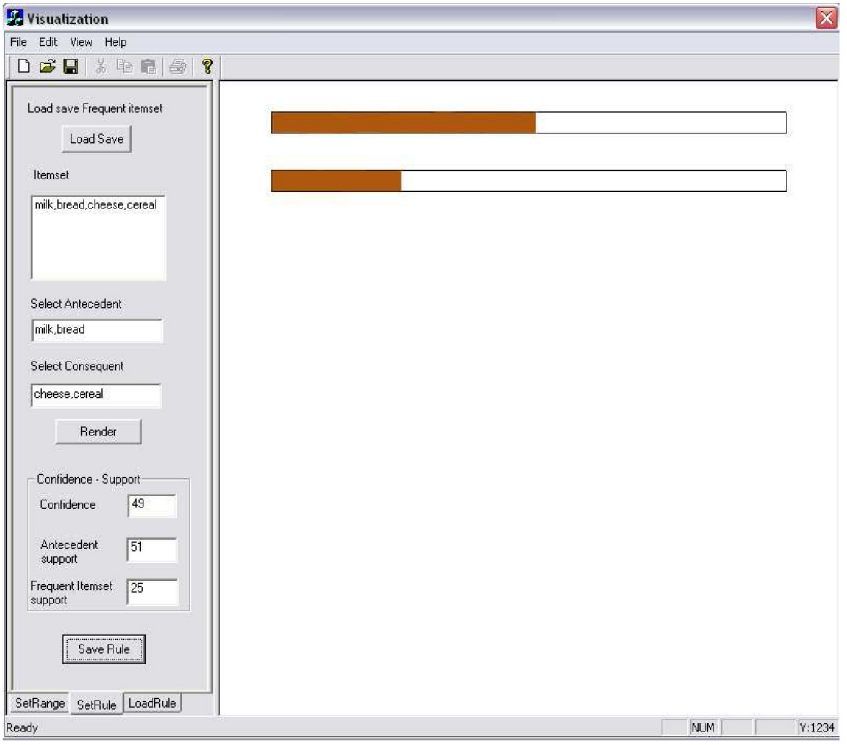


Fig. 2. The right drawing space represents two barsticks. The first bar shows the proportion of antecedent of the association rule. The second bar shows the consequent based on the selected antecedent. The control tab on top of the left hand side is to input the antecedent and consequent of the rule. The bottom of the tab displays the confidence, the antecedent support, and the itemset support.

4 Conclusion and Future Work

In most visualization tools for association rule mining, the visualization and association rule mining tend to be implemented seperately. Association rule mining algorithms generate all frequent itemsets and association rules. The visualization approaches are only used to visualize the association rules obtained from the mining algorithm. Our technique is different from these techniques. The visualization in our system is integrated in the mining algorithm to incorporate human knowledge into the mining process. One drawback of automatic mining algorithms is that all frequent itemsets and association rules generated may not be interesting and other techniques are needed to find the interesting rules. This may itself be a time-consuming task. However, in our technique, the user has complete control to choose the frequent itemset and association rules. As a result, the user can control the mining process starting from choosing a frequent itemset, choosing the antecedents and consequents of the association rules and finally visualizing

the mined rules. The main advantage of our technique is that it can avoid generating many uninteresting association rules.

Our technique in completely scalable in terms of its ability to visualize number of data items. The number of records each vertical line in a bar stick represents depends on the total number of records in the database. While each vertical line represents only one record in the simplest case, there is no upper limit of records that it can represent for a large database. We are currently integrating our visualization tool with MS Access relational database through ODBC. We execute queries to retrieve complete columns from tables in MS Access and use these columns to display records in our visualization tool. We have been able to load data from tables with upto 100,000 records. We expect that our tool will be part of an integrated environment for visual data mining.

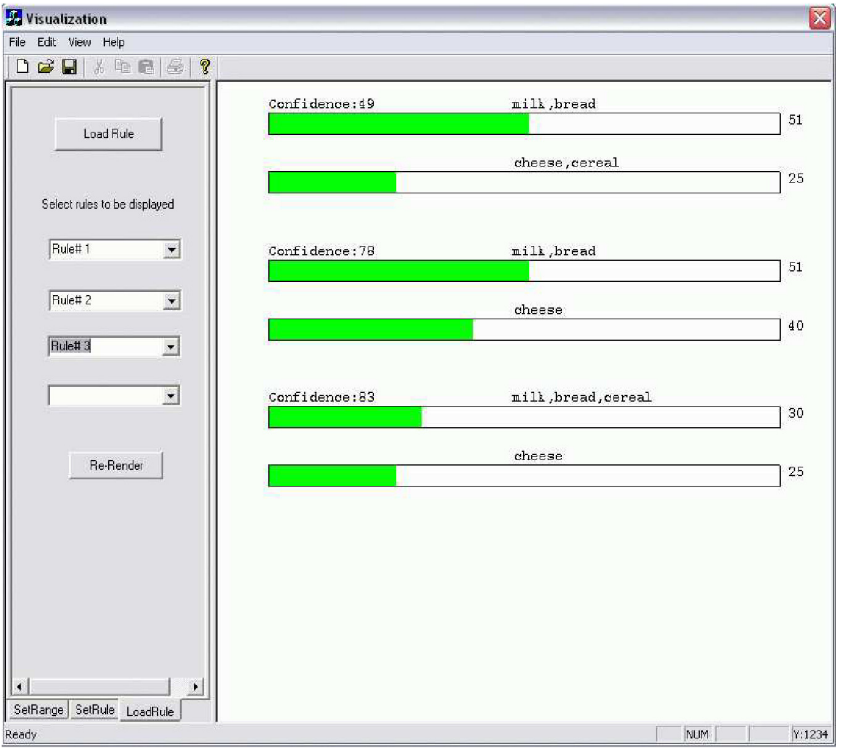


Fig. 3. Illustration for deriving interesting association rules from the selection of the rules in Figure 2. The two bars and the texts represent each rule and its properties.

Acknowledgments. The second author's research is partially supported by Western Australian Interactive Virtual Environments Centre (IVEC) and Australian Partnership in Advanced Computing (APAC).

References

1. R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In J. B. Bocca, M. Jarke, and C. Zaniolo, editors, *VLDB'94, Proceedings of 20th International Conference on Very Large Data Bases, September 12-15, 1994, Santiago de Chile, Chile*, pages 487–499. Morgan Kaufmann, 1994.
2. W. S. Cleveland. Visualizing data. *Hobart Press Summit*, 1993.
3. U. Fayyad, G. G. Grinstein, and A. Wierse. *Information Visualization in Data Mining and Knowledge Discovery*. Morgan Kaufmann, 2002.
4. S. K. Feiner and C. Beshers. Worlds within worlds: Metaphors for exploring n-dimensional virtual worlds. In Scott E. Hudson, editor, *User interface software and technology*. ACM Press, October 1990.
5. J. Han and M. Kamber. *Data Mining Concepts and Techniques*. Morgan Kaufmann, 2001.
6. H. Hofmann, A.P. Siebes, and A.F. Wilhelm. Visualizing association rules with interactive mosaic plots. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 227–235. ACM Press, 2000.
7. SAS Institute Inc. `http://www.sas.com/technologies/analytics/datamining/miner/`.
8. A. Inselberg and B. Dimsdale. Parallel coordinates for visualizing multidimensional geometry. *Computer Graphics (Proceedings of CG International)*, pages 25–44, 1987.
9. D.A. Keim and H.P. Kriegel. Visdb: database exploration using multidimensional visualization. *IEEE Computer Graphics and Applications*, 14:40–49, 1994.
10. K-H. Ong, K-L. Ong, W-K. Ng, and E-P. Lim. Crystalclear: Active visualization of association rules. In *International Workshop on Active Mining (AM-2002), in conjunction with IEEE International Conference On Data Mining, Maebashi City, Japan, 9 December 2002*.
11. A. Savasere, E. Omiecinski, and S. B. Navathe. An efficient algorithm for mining association rules in large databases. In U. Dayal, P. M. D. Gray, and S. Nishio, editors, *VLDB'95, Proceedings of 21th International Conference on Very Large Data Bases, September 11-15, 1995, Zurich, Switzerland*, pages 432–444. Morgan Kaufmann, 1995.
12. SGI. `http://www.sgi.com/software/mineset.html`.
13. R. Srikant and R. Agrawal. Mining generalized association rules. In U. Dayal, P. M. D. Gray, and S. Nishio, editors, *VLDB'95, Proceedings of 21th International Conference on Very Large Data Bases, September 11-15, 1995, Zurich, Switzerland*, pages 407–419. Morgan Kaufmann, 1995.
14. C. Stolte, D. Tang, and P. Hanrahan. Polaris: a system for query, analysis, and visualization of multidimensional relational databases. *IEEE Transactions on Visualization and Computer Graphics*, 8:52–65, 2002.
15. K. Techapichetvanich, A. Datta, and R. Owens. Hddv: Hierarchical dynamic dimensional visualization. In *Proc. IASTED International Conference on Databases and Applications Innsbruck, Austria, February 2004, to appear*.
16. K. Wang, Y. Jiang, and L. V. S. Lakshmanan. Mining unexpected rules by pushing user dynamics. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 246–255. ACM Press, 2003.
17. P. C. Wong, P. Whitney, and J. Thomas. Visualizing association rules for text mining. In *INFOVIS*, pages 120–123, 1999.

Visualizing Predictive Models in Decision Tree Generation

Sung Baik[1], Jerzy Bala[2], and Sung Ahn[3]

[1] Sejong University, Seoul 143-747, KOREA
sbaik@sejong.ac.kr
[2] Datamat Systems Research, Inc.
1600 International Drive, McLean, VA 22102, USA
jbala@dsri.edu
[3] Kookmin University, Seoul 136-702, KOREA
sahn@kookmin.ac.kr

Abstract. This paper discusses a visualization technique integrated with inductive generalization. The technique represents classification rules inferred from data, as landscapes of graphical objects in a 3D visualization space, which can provide valuable insights into knowledge discovery and model-building processes. Such visual organization of classification rules can contribute to additional human insights into classification models that are hard to attain using traditional displays. It also includes navigational locomotion and high interactivity to facilitate the interpretation and comparison of results obtained in various classification scenarios. This is especially apparent for large rule sets where browsing through textual syntax of thousands of rules is beyond human comprehension. Visualization of both knowledge and data aids in assessing data quality and provides the capability for data cleansing.

1 Introduction

Information visualization in the knowledge discovery process enables humans to understand easily and intuitively solutions, of the given problems, inferred from huge amounts of data. It can also release the data analyst's cognitive burden in the process of data mining since it requires users to understand complex data formats and knowledge representations in the process of discovering and interpreting new patterns [1,2]. An example of the cognitive burden to humans is the use of textual representations to force the human analyst into non-optimal modes of information processing (e.g., requiring high demands on those aspects of cognitive functions that are most limited in humans such as high engagement of short term and long term memory in the process of remembering and comparing textual syntaxes).

Synergistic integration of traditional analytical methods with visual presentation techniques can enhance the effectiveness of overall data mining process [3]. Visualization techniques in each phase of data mining are summarized as follows:

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 489–495, 2004.
© Springer-Verlag Berlin Heidelberg 2004

- Visualization in data preparation: This technique helps analysts to manipulate data for data transformation, sampling, data selection, handling missing data fields in a visual way.

- Visualization in model construction: For model derivation, users need to know the prior information for model construction. (e.g., they select the training set and define parameters of models.) This technique guides them to carry out model construction through the interactive visualization user interface [4,5].

- Visualization in model understanding and validation: The important advantage of visualization is that it plays to the strengths of human cognition. From a human perspective, visualization applied to information extraction and knowledge discovery offers a powerful means of analysis that can help uncover patterns and trends that are likely to be missed with other non-visual methods. Most importantly, visual presentations enable humans to make observations without pre-conception (i.e., the visual presentation shows what is important) by taking advantage of the strengths of human cognitive abilities. Visualization allows auditing, intimate involvement, and the deeper understanding of data/knowledge sets.

Several representations such as decision tree, decision rule, production rule, and decision graph are used for visualization of classification data with graphic form. They can enhance the user's capabilities to see, explore, and gain decision-making insights. Researcher's selection of such representations with visualization depends upon how well humans can understand the knowledge inferred from the given data [6]. This paper focuses on the visualization of the predictive models of decision rules in decision tree generation through the transfer of data mining and decision support processes to the visualization space. The user can interact with the system through the visual representation space where various graphical objects are rendered. Graphical objects represent data, knowledge (e.g. as induced rules), and query explanations (i.e., decisions on unknown data identifications). The system integrates graphical objects through the use of visually cognitive, human oriented depictions. A user can also examine non-graphical explanations (i.e. text based) to posed queries.

2 Knowledge/Data Visualization with Data Mining

This paper discusses a robust data mining system [7] combining knowledge/data visualization by comparing with decision tree classification [8] of SAS Enterprise Miner (E-miner), that is a complete data mining solution. Whereas E-miner provides a simple visualization of decision tree to represent classification results, the data mining system provides visually cognitive, human oriented depictions to humans for better understanding. The system consists of three following major components:

1. **Classification:** This component generates classification rules using a decision tree learning mechanism. The tree generation process is based on the Likelihood-based Model Evaluation algorithms. It also utilizes algorithms for attribute quanti-

zation, missing value handling, advanced model refinement, model based data cleansing, and model boosting.

2. **Prediction:** This component uses induced rules to predict class memberships of queried data (i.e., usually previously unknown) and generates statistics (e.g., degree of match to each rule, degree of match to a class, etc.).
3. **Knowledge/Data Visualization:** This component provides a 3D Graphical User Interface with graphical objects representing classification rules (e.g., decision rules induced from known sample data), graphically depicted responses to user's queries (e.g., prediction results on unknown data), and visualization of the data used to generate rules. Properties of these objects (e.g., size and color) represent and/or are correlated to various rule statistics (e.g., number of data points covered by a rule, a rule firing rate, etc.) A set of such visualized rules forms the classification landscape. The user can navigate the landscape by zooming, rotation, and translation, inspect visualized rules via "semantic zooming" (i.e. displaying their syntax), and render a 3D plot of data linked to a given rule. Landscapes can also be queried for predicting class memberships of unknown data and reporting prediction results visually (e.g., changing the color of a firing rule).

3 Classification Landscape

Using GUI controls and a mouse, the following operations can be executed:

1. Landscape Navigation: The user can change views by zooming on the landscape, rotating around it, and/or translating the landscape display.
2. Semantic Zooming: A semantic zooming operation (i.e., display of rule syntax) can be performed by *brushing* on a given sphere with the mouse pointer.
3. Querying: The landscape can be queried to predict the class memberships of unknown data. A firing rule changes its color intensity proportionally to a percentage of data points matching this rule.
4. Generation of Multiple Views: Multiple landscapes can be rendered in the visualization space.
5. Visual Pruning: The user can remove from the landscape those rules that do not satisfy some threshold condition, e.g, number of data points covered.
6. Linking to Data: Moving the mouse pointer to a sphere and clicking on it invokes the process of displaying 3D plots of data points covered by the associated rule. This process can be used for data inspection and cleansing.

The following two modes of operation are available through the classification landscape:

- Knowledge Discovery: This mode uses the inference engine module to generate classification landscapes to be observed, navigated, and inspected by the user.
- Decision Support: In this mode, the predictor module is used to query unknown data for its class membership using classification landscapes.

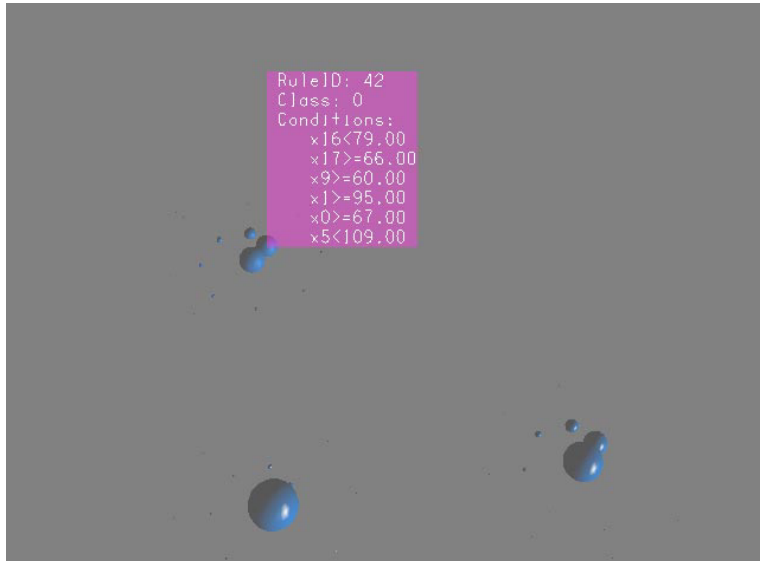


Fig. 1. A Three-Cluster Classification Landscape from a remote-sensing database

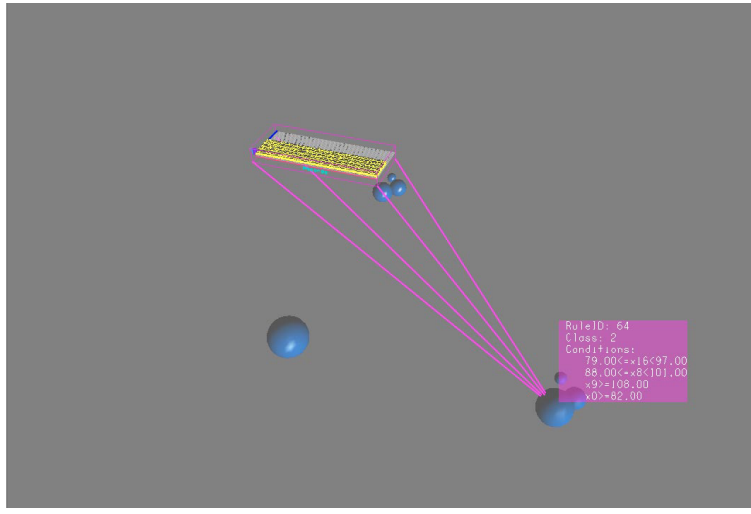


Fig. 2. A Three-Cluster Classification Landscape after visual pruning in Figure 1 with Linking to Data

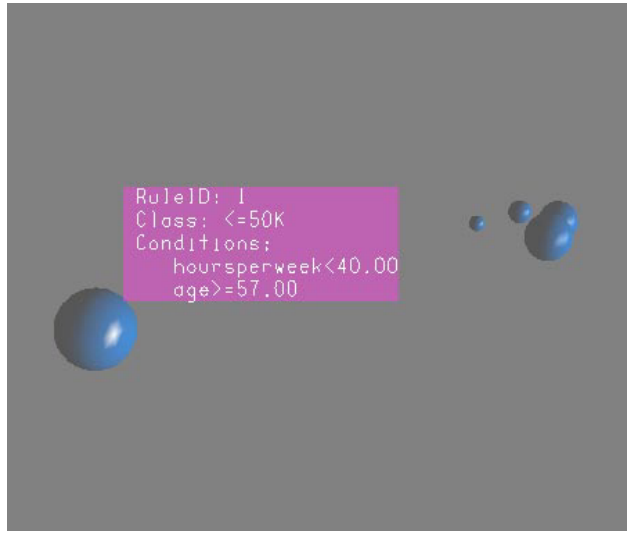


Fig. 3. An example of Two-Class Landscape from a census-income database

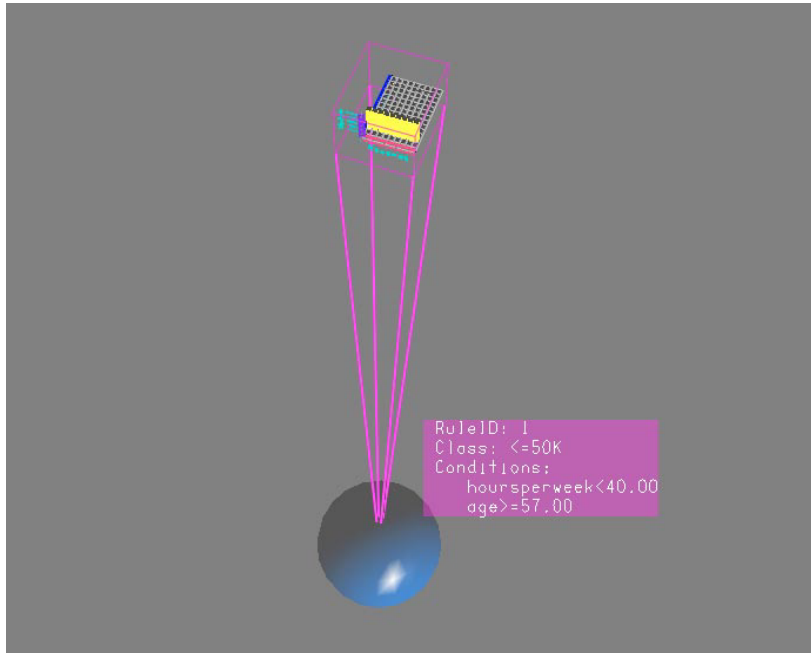


Fig. 4. An example of Linking to Data from a census-income database

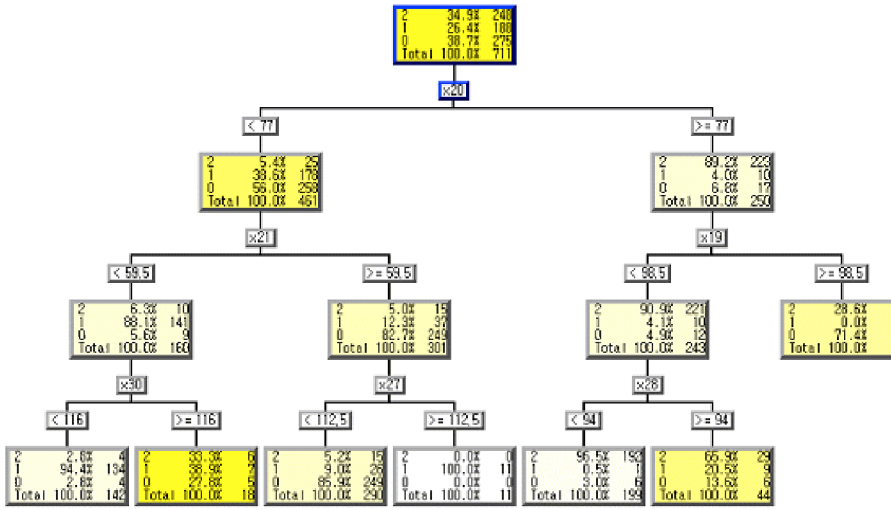


Fig. 5. The classification result represented by a decision tree provided by E-miner from a remote-sensing database

4 Examples of Classification Landscape

We have two kinds of data for classification scenarios to generate patterns in the form of classification rules that can lead to discovery of unknown information:

1. Census-income database [9]: It contains weighted census data extracted from the 1994 and 1995 current population surveys conducted by the U.S. Census Bureau. The data contains demographic and employment related fields.
2. Australian Center for Remote Sensing database: It consists of the multi-spectral values of pixels in 3x3 neighborhoods, and the classification associated with the central pixel in each neighborhood. The data associated with each spectral band reside in logically different databases.

Figure 1 and 2 depicts an example of the classification landscape for decision rules inferred from a remote-sensing database. The landscape consists of three spiral-formed clusters of spheres representing two different classes. The center of each cluster is populated with the most dominant rules (larger spheres), i.e., the rules that cover large number of data points. The sphere size is proportional to the number of data points covered by the rule it represents. Semantic zooming is depicted as a rule textual syntax in a boxed area. Figure 2 represents the classification landscape after visual pruning and includes raw data linked with a specific rule. Figure 3 depicts a two-class classification landscapes for decision rules inferred from the census-income database, respectively. Figure 4 shows rendering of data points associated with a given rule in Figure 3. Figure 5 is a classification result represented by a decision tree provided by E-miner from the remote-sensing database.

References

1. Hao, M.C.; Dayal and U., Hsu, M., A Java-based visual mining infrastructure and applications, Proceedings of 1999 IEEE Symposium on Information Visualization, 24-29, pp124-127, 1999
2. Robinson, N.and Shapcott, M., Data mining information visualisation - beyond charts and graphs, Proceedings of Sixth International Conference on Information Visualisation, 10-12, pp577-583, 2002
3. Kopanakis, I. and Theodoulidis, B., Visual data mining modeling techniques for the visualization of mining outcomes, Journal of Visual Languages & Computing, 14-6, pp543-589, 2003
4. Manco, G., Pizzuti, C. and Talia, D., Eureka!: an interactive and visual knowledge discovery tool, Journal of Visual Languages & Computing, 15-1, pp1-35, 2004
5. Ware, M. *et al.*, Interactive machine learning: letting users build classifiers, International Journal of Human-Computer Studies, 55-3, pp281-292, 2001
6. Humphrey, M., Cunningham, S. and Witten, I., Knowledge Visualization Techniques for Machine Learning, Intelligent Data Analysis, 2-1, pp333-347, 1998
7. Bala, J., Baik, S., Gutta, S., Hadjarian, A., Mannucci, M., and Pachowicz, P., InferView: An Integrated System for Knowledge Acquisition and Visualization," proceedings of the Federal Data Mining Symposium and Exposition 99, 1999
8. See Web site at http://www.sas.com/technologies/analytics/datamining/miner/dec_trees.html
9. See Web site at http://kdd.ics.uci.edu/databases/census-income/census-income.data.html

A Model for Use Case Priorization Using Criticality Analysis

José Daniel García, Jesús Carretero, José María Pérez, and Félix García

Computer Architecture Group,
Universidad Carlos III de Madrid,
Avda. de la Universidad Carlos III, 22,
28270 Colmenarejo, Madrid, Spain
Fax: +34 91 8561270
jdaniel@arcos.inf.uc3m.es
http://www.arcos.inf.uc3m.es

Abstract. Modern UML based software development processes have use cases specification as a keystone of functional requirement modeling. Use cases have been used with success in broader contexts than software development as systems engineering. Not all the use cases are equally critical for the system to satisfy its goals. Criticality analysis allows to identify the critical elements of a system. Applying criticality analysis to functional requirements of a system allows to identify which are the critical functionalities of a system. In this paper we present a use case model structuring scheme. We apply a method to evaluate the criticality of use cases of a system to our structuring model. We also propose the requirements for an UML profile for criticality analysis of systems.

1 Introduction

Use cases have been used as a mean of capturing the requirements of a software system by several methods [1,2,3]. Use cases have also been used with success in broader contexts than software development as systems engineering [4,5,6].

When modeling functional requirements as use cases, those requirements are represented as a set of use cases, where each use case is the specification of a set of transactions between the system and the external actors, which yield an observable result that is, typically, of value for one or more actors or other stakeholders of the system [7].

Use cases are an unstructured mechanism for representing functional requirements of a system. However some structuring mechanisms have been proposed such as the use of goals [8,9,10,11,12,13], the grouping of use cases following some clustering criteria, into use case packages [2,12] or the partitioning of the system into smaller collaborating subsystems [14].

Besides, UML offers two standard mechanisms for structuring use cases: *extension* and *inclusion*. These mechanisms may be used to share common behavior from different use cases [7].

Usually, use cases are prioritized using a subjective ranking [15] or mental perception of the stakeholders views [16,17]. Another approach is to rank use

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 496–505, 2004.
© Springer-Verlag Berlin Heidelberg 2004

cases according with their criticality impact on the whole system. Criticality based ranking allows deciding which functionality is critical for the system and taking special measures (quality assurance, fault tolerance, testing coverage) on it.

Performing a criticality analysis on use cases is the first step to determine which are the most critical components of a system when using the reliability centered maintenance (RCM) approach [18]. RCM [19,20] is a set of techniques which have been applied in several industrial fields such as aviation [21], oil industry [22] and ships [23]. RCM can be defined as a systematic approach to systems functionality, failures of that functionality, causes and effects of failures and infrastructure affected by failures. The key objective of RCM is to maintain a system in a *'good'* functional state.

This paper is organized as follows: section 2 presents sample factors having impact on the criticality of a use case; section 3 gives a model to compute the criticality of a use case; section 4 shows how to perform a criticality analysis over a use case model; section 5 shows the extension requirements to the use case metamodel to support criticality analysis; section 6 presents the obtained conclusions; and section 7 establishes the lines for a future work.

2 Criticality Factors for a Use Case

Functional requirements of a system may be described as a set of use cases. When evaluating the criticality of a use case, several factors must be taken into account. Each criticality factor may be seen as a dimension of criticality.

In [18] the following factors are proposed:

- **Safety**: This factor evaluates the required safety level for the functionality. Use cases needing to provide a high safety level are more critical.
- **Security**: This factor evaluates the impact of the required security level. Use cases needing to provide a higher security level are more critical.
- **Infrastructure**: This factor evaluates the impact of hardware resources (computing nodes, communication infrastructure, devices). As the dependency of a use case on infrastructure elements increases, its criticality also increases.
- **Frequency**: This factor evaluates the execution frequency of the use case. More frequently executed use cases are more critical than rarely executed use cases.
- **Revenues**: This factor evaluates the revenues obtained or expected thanks to the use case. Use cases participating in high revenue processes are more critical than those giving low revenues.
- **Availability**: This factor evaluates the required availability of the functionality provided by the use case. Use cases that must be available permanently are more critical.
- **Reliability**: This factor evaluates the required reliability of the use case. Use cases with strong reliability requirements are more critical than the rest of use cases.

These factors are only an example of a set of factors.

3 Criticality Computation for a Use Case

Functional requirements of a system may be described as a set of use cases (see equation 1) where each use case u_i belongs to the set R of use cases.

$$R = \{u_i\}_{i=1}^{n} \tag{1}$$

Thus, for a system whose functionality is modeled by n use cases, its R set is $\{u_1, u_2, \ldots u_n\}$.

Let u_i be a use case. Criticality of use case u_i (noted by $c(u_i)$) is a measure of the relative importance of use case u_i for the system, from a functional point of view. Criticality values may be expressed as real numbers. The higher the criticality is, the more critical the corresponding use case is. Thus, criticality may be seen as function mapping every use case to real numbers set as expressed in 2.

$$c : R \mapsto \Re \tag{2}$$

For every use case u its criticality is computed by the function $c(u)$. Function $c(u)$ may be defined in several ways. For example, $c(u)$ may be computed by taking into account the criticality levels for a set of criticality factors.

We define $l_j(u_i)$ as the criticality level of factor j for use case u_i. In this way, a criticality level may be evaluated independently for each factor. Each criticality level function l_j is a function from the use case set R in the real numbers set:

$$l_j : R \mapsto \Re \tag{3}$$

For the evaluation of each of this functions $l_j(u_i)$ expert knowledge may be used. Not all these functions will be equal, as the evaluation will depend on the own characteristics of the evaluated factor.

Let L the set of all the criticality level functions:

$$L = \{l_1(u), l_2(u), \ldots, l_m(u)\} \tag{4}$$

Criticality function $c(u)$ may be defined, in its general form, as any transformation $g(L)$ on set of functions L giving a real number.

$$c(u) = g(l_1(u), l_2(u), \ldots, l_m(u)) \tag{5}$$

A first approach for computing criticality could be to compute the sum of all the l functions as presented in equation 6.

$$c(u) = \sum_{k=1}^{m} l_k(u) \tag{6}$$

However, in most cases, importance of each factor is not the same. This may be imposed by company or regulation entities policies or may be derived from the project context. To accommodate this fact, we have introduced a weight for each factor in the formula and have normalized weight values.

$$c(u_i) = \sum_{k=1}^{m} \frac{w_k}{\sum_{j=1}^{n} w_j} \cdot l_k(u_i) \tag{7}$$

Some definitions may be made to simplify the notation:

$$w_k^* = \frac{w_k}{\sum_{j=1}^{m} w_j} \tag{8}$$

Giving a simplified notation for the criticality function on a use case:

$$c(u_i) = \sum_{k=1}^{m} w_k^* \cdot l_k(u_i) \tag{9}$$

Where w_k is the normalized weight of the kth criticality factor and $l_k(u_i)$ is the criticality level of that factor for the computed use case.

4 Criticality Analysis of a Use Case Model

Requirements for a simple system may be expressed as a set of use cases. In such situation, criticality analysis may be applied to each use case as described in section 3. But when the number of use cases increases this approach is not effective because of two different reasons:

- Having a plain use case model for more than 80-100 use cases makes it very difficult to manage, so that some structuring technique is needed to reduce its complexity [12].
- As the number of use cases increases, the effort needed to perform a criticality analysis at the use case level also increases. The main objective of criticality analysis at the use case level is to identify the most critical use cases for the system, so analyzing every use case is not cost effective for a large system. The concept of criticality inheritance [24] may help here.

This section studies how use case criticality analysis may be performed in the presence of a structured use case model. First, criticality analysis for every structuring mechanism is studied. Then, a general method for criticality analysis of use case models is proposed.

4.1 Structuring Mechanisms

Inclusion Relationship. An <<*include*>> relationship between two use cases means that *the behavior defined in the including use case is included in the behavior of the base use case* [7]. That is, a base use case explicitly incorporates the behavior of another use case at a location specified in the base.

The include relationship is intended to be used when there are common parts of the behavior of two or more use cases. This common part is then extracted to a separate use case, to be included by all the base use cases having this part in common.

Included use cases are not full use cases in the sense that an included use case instance cannot happen by itself, but in the context of a base use case. In this sense an included use case may be classified as a subfunction [12].

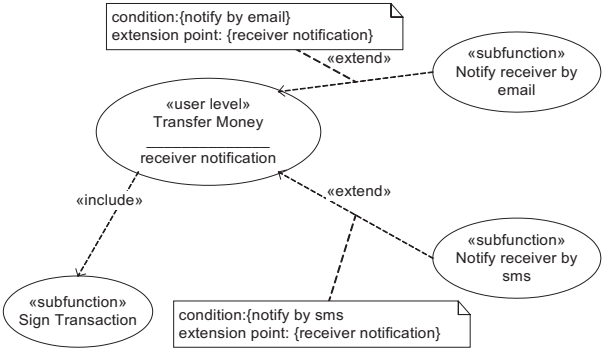


Fig. 1. Use case <<*include*>> and <<*extend*>> relationships

In our use cases we use the stereotype <<*subfunction*>> for the use cases which are included by others which cannot be instantiated by themselves. A typical example of <<*subfunction*>> use case is *sign transaction*. We use the stereotype <<*user level*>> for use cases which can be instantiated such as *transfer funds*. Figure 1 shows the use cases *transfer funds* and *sign transaction*.

Let u_i be a <<*user level*>> use case including a subset of <<*subfunction*>> use cases $\{f_1 \dots f_k\}$. In such a case, criticality of u_i will always be an upper bound for criticality of each included use case f_i. That is:

$$c(u_i) \geq max_{i=1}^{k} \{c(f_i)\} \tag{10}$$

Extension Relationship. An <<*extends*>> relationship specifies that *the behavior of a use case may be extended by the behavior of another (usually supplementary) use case* [7].

The extension takes place at one or more specific extension points defined in the base use case. However, the base use case is defined independently of the extending use case and is meaningful independently of it. The base use case defines its extension points where the behavior of the extending use case may happen optionally.

Note that the same extending use case may extend more than one base use case. Furthermore, an extending use case may itself be extended.

Extending use cases typically define behavior that is not meaningful by itself but in the context of the base use cases it extends. An extending use case defines a modular behavior increment that augments an execution of the extended use case under specific conditions.

For example, we may have a base use case such as *transfer funds*. That use case may be extended under the condition *notify by email* so that additional functionality *notify receiver by email* is used. The same use case may be extended under the condition *notify by sms* so that additional functionality *notify receiver by sms* is used. Figure 1 shows use cases *transfer funds*, *notify receiver by email* and *notify receiver by sms*.

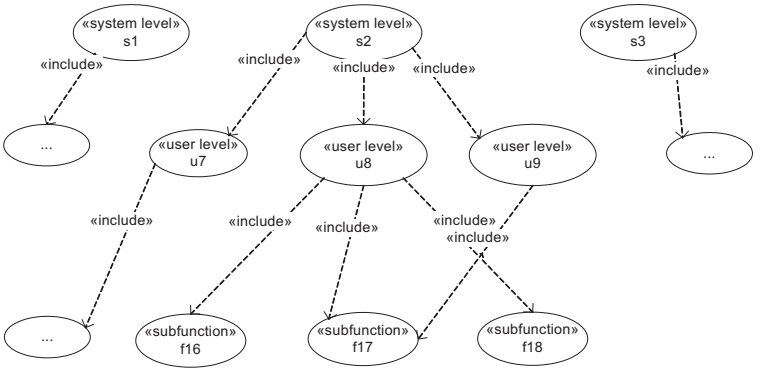


Fig. 2. Use case structure with <<*system level*>>, <<*user level*>> and <<*subfunction*>> use cases

Let u_i be a <<*user level*>> use case which is extended by a set of <<*user level*>> use cases $\{e_1 \dots e_k\}$. In such a case, criticality of u_i will always follow:

$$c(u_i) \leq max_{i=1}^{k} \{c(e_i)\} \tag{11}$$

System Use Cases. All use cases related to an specific function may be grouped to allow complexity management. One way of structuring use cases is to define a higher level of use cases. These uses cases are not at the user level, but at the system (as a whole) level.

In our use cases we use the stereotype <<*system level*>> for system use cases. Each system level use case concentrates in achieving a goal of the system.

Relationships among <<*system level*>> use cases and <<*user level*>> use cases have not been yet fully investigated. In general [12], an <<*include*>> relationship is suggested, so that the set of <<*system level*>> use cases is a partition of the set of <<*user level*>> use cases (i.e. a <<*user level*>> use case is included only by one <<*system level*>> use case).

Figure 2 shows a typical use case structured model.

Let s_i be a system level use case which includes a set of user level use cases $\{u_1 \dots u_k\}$. In such a case, criticality of s_i will always follow:

$$c(s_i) \geq max_{i=1}^{k} \{c(u_i)\} \tag{12}$$

4.2 Criticality Analysis of Structured Use Case Models

As it has already been noted, criticality analysis of a complex use case model is not cost effective, as the effort needed to perform the analysis starts to be very high. The objective must be to find the most critical use cases without evaluating criticality for every use case of the system. In such cases, having a structured use case model may help by allowing the application of the criticality inheritance concept [24].

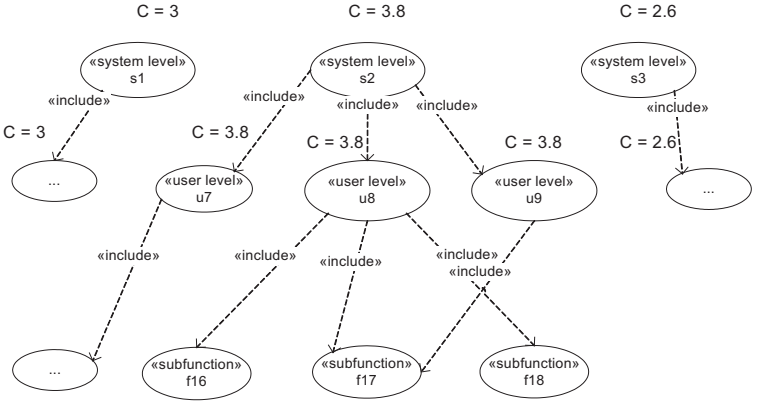


Fig. 3. Criticality inheritance after <<*system level*>> criticality analysis

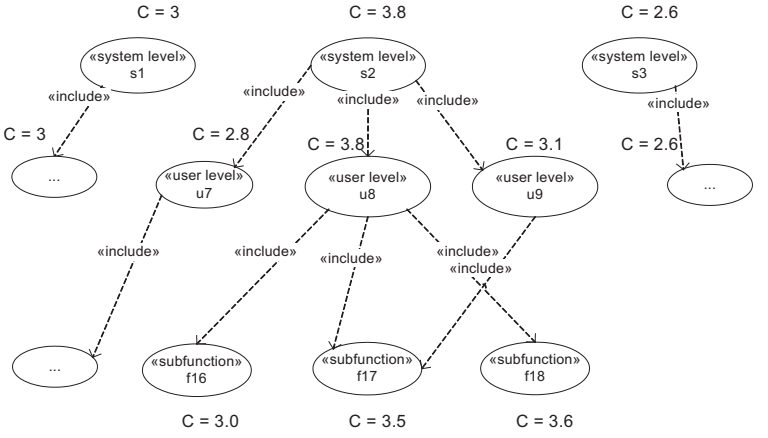


Fig. 4. Criticality inheritance after completing criticality analysis

Criticality is first computed for the <<*system level*>> use cases, and then inherited (see figure 3) in a top-down manner. In this way, every use case starts having a criticality estimated value which is an upper bound for its actual criticality value.

After this first step, we perform a criticality analysis for <<*user level*>> use cases being included by most critical <<*system level*>> use cases (see figure 4). This process is repeated recursively taking as input for each iteration most critical use cases.

At the end of this process we have as a result ranked list of use cases in which the ranking criteria is criticality. This information is very valuable for the rest of the project and may be used in several ways.

- Very critical use cases may be analyzed to determine if there is some measure to reduce its criticality. For example, it could be evaluated if availability could be reduced.
- During design, criticality information may be used to specially design most critical use cases. For example, include more inspections for the design of critical use cases.
- Criticality information may be used to decide which use cases are implemented in a fault tolerant way.
- Criticality information may be used at testing to assign testing efforts attending to criticality. In this way, most critical components will be tested more.

5 Extension of Use Case Metamodel for Criticality Analysis

To support the criticality analysis on use case model, the following requirements must be satisfied:

- A structuring mechanism for use case modeling must be available. We propose to do so by stereotyping use cases.
- Each use case must be able to have associated values for the criticality level of each factor. We propose to do so through the use of tagged values.
- Each use case must be able to have an associated criticality value. We propose to do so through the use of a tagged value.

So our proposal defines three stereotypes for use cases:

- <<*system level*>>: To be applied to system level use cases.
- <<*user level*>>: To be applied to user level use cases.
- <<*subfunction*>>: To be applied to subfunction use cases.

These stereotypes will support the following tagged values:

- *criticality factors*: A set of real numbers (one for each criticality factor).
- *criticality value*: A real number.

Given a use case u_i at any level, the criticality value is computed as follows:

1. If the factor levels are unknown, the criticality is inherited from the upper level.
2. If the factor levels are known the criticality is computed as:

$$c(u_i) = max(\sum_{k=1}^{n} w_k l_k(u_i),\ \{c(u_j)\ \forall u_i\ include\ u_j\}) \tag{13}$$

6 Conclusion

In this paper we have presented a method for computing the criticality of an individual use case based on the criticality levels of a set of factors.

We have also extended our criticality analysis method for complex systems consisting of a large number of use cases, considering the impact of use cases standard relationships and the use of a structured use case model.

Our third contribution has been an extension of the UML use case metamodel to support criticality analysis. The extension proposes the use of three stereotypes for use cases and a set of tagged values to represent the criticality properties of each use case.

7 Future Work

Having performed a criticality analysis at the functional requirements level, this information may be propagated and refined to other modeling artifacts. We plan to extend our criticality analysis model to the rest of UML artifacts so that criticality may be known for every element (hardware or software) of a system.

We will integrate this work in a *UML profile for Criticality Analysis of Systems*.

References

1. Jacobson, I.: Software Engineering - A Use Case Driven Approach. Addison-Wesley (1992)
2. Jacobson, I., Booch, G., Rumbaugh, J.: The Unified Software Development Process. The Object Technology Series. Addison-Wesley (1999)
3. D'Souza, D.F., Wills, A.C.: Objects, Components and Frameworks with UML. The Catalysis Approach. The Object Technology Series. Addison-Wesley (1998)
4. Alexander, I., Zink, T.: Introduction to systems engineering with use cases. Computing and Control Engineering Journal **13** (2002) 289–297
5. Krikorian, H.F.: Introduction to object-oriented systems engineering, part 2. IT Professional **5** (2003) 38–42
6. Krikorian, H.F.: Introduction to object-oriented systems engineering, part 2. IT Professional **5** (2003) 49–55
7. Object Management Group: UML 2.0 Superstructure Specification. (2003) Final Adopted Specification.
8. Cockburn, A.: Goals and use cases. Journal on Object Oriented Programming **10** (1997) 35–40 http://members.aol.com/acockburn/papers/usecases.htm.
9. Cockburn, A.: Using goal-based use cases. Journal on Object Oriented Programming **10** (1997) 56–62 http://members.aol.com/acockburn/papers/usecases.htm.
10. Lee, J., Xue, N.L.: Analyzing user requirements by use cases: A goal-driven approach. IEEE Software **16** (1999) 92–101
11. Mylopoulos, J., Chung, L., Yu, E.: From object-oriented to goal-oriented requirements analysis. Communications of the ACM **42** (1999) 31–37
12. Cockburn, A.: Writing Effective Use Cases. The Agile Software Development Series. Addison-Wesley (2001)

13. Lee, J., Xue, N.L., Kuo, J.Y.: Structuring requirement specifications with goals. Information and Software Technology **43** (2001) 121–135
14. Nasr, E., mcDermid, J., Bernat, G.: A technique for managing complexity of use cases for large complex embedded systems. In: Proceedings of the Fifth IEEE International Symposium on Object-Oriented Real-Time Distributed Computing. (2002)
15. Thelin, T., Runeson, P., Wohlin, C.: Prioritized use cases as a vehicle for software inspections. IEEE Software **20** (2003) 30–33
16. Moisiadis, F.: Prioritising scenario evolution. In: Proceedings of the 4th International Conference on Requirements Engineering, IEEE (2000) 85–94
17. Moisiadis, F.: Prioritising use cases and scenarios. In: Proceedings of the 37th International Conference on Technology of Object-Oriented Languages and Systems. TOOLS-pacific 2000, IEEE (2000) 108–119
18. García, J.D., Pérez, J.M., Carretero, J., García-Carballeira, F.: Reducing software maintenance cost using reliability centered maintenance (RCM) and expert knowledge. In: Concurrent Engineering. Advanced Design, Production and Management Systems. Volume 2., International Society for Productivity Enhacement, A.A. Balkema (2003) 379–385 ISBN: 90-5809-234-6.
19. Moubray, J.: Reliability Centered Maintenance. 2nd edition edn. Industrial Press Inc. (1997)
20. Rausand, M.: Reliability centered maintenance. Reliability Engineering and System Safety **60** (1998) 121–132
21. Naval Air Systems Command, USA: Guidelines for the Naval Aviation Reliability-Centered Maintenance. (1996)
22. OREDA Consortium: Reliability Data Handbook. 4th edition edn. (2003)
23. Shculkins, N.: Application of the reliability centered maintenance structures methods to ships and submarines. Maintenance **11** (1996)
24. Carretero, J., Pérez, J.M., García-Carballeira, F., Calderón, A., Fernández, J., García, J.D., Lozano, A., Cardona, L., Cotaina, N., Prete, P.: Applying RCM in large scale systems: a case study with railway networks. Reliability Engineering and System Safety **82** (2003) 257–273

Using a Goal-Refinement Tree to Obtain and Refine Organizational Requirements*

Hugo Estrada[1,2], Oscar Pastor[1], Alicia Martínez[1,3], and Jose Torres-Jimenez[4]

[1] Technical University of Valencia
Avenida de los Naranjos s/n, Valencia, Spain
{alimartin, opastor, hestrada}@dsic.upv.es
[2]CENIDET Cuernavaca, Morelos, Mexico
[3] I.T. Zacatepec, Morelos, Mexico
[4]ITESM Campus Cuernavaca, Morelos, Mexico, jtj@itesm.mx

Abstract. At present, the organizational requirements are considered to be one of the most important aspects in the development of information systems. Many research efforts in software engineering have focused on integrating organizational modeling as a key piece in requirements engineering. However, the majority of these works focus only on the definition of notations that permit the representation of the semantics of the organizational context, and only a few works define processes that use these notations in a methodological way. This lack of a methodological process for generating organizational models makes practical application in software development enterprises difficult. The objective of this paper is to present a goal-based method to obtain and refine organizational requirements. These requirements are used to validate the understanding of the organizational process before constructing the information system. This will enable us to develop information systems that integrate the necessary functionality so that the organizational actors perform their tasks and fulfill their goals.

1 Introduction

In recent years, many efforts have been made to define a software production process which is precise, rigorous and reliable and where the resulting information system clearly satisfies the users' needs [1], [2], [3], [4], [5]. The majority of these works use the system requirements as a starting point. Even if this resolves many of the problems related to the generation of the software product, it doesn't assure that it represents the functionality expected by the organizational users to perform their particular goal and the general goal of the enterprise. In these production processes, there is one main feature that is not properly taken into account: the importance of understanding that the information system should be the correct representation of the requirements taken from the organizational model. It is only possible to generate a

* This work has been partially supported by the MCYT project with ref. TIC2001-3530-C02-01, the Technical University of Valencia, Spain and the National Association of Universities and Institutions of Higher Education ANUIES, Mexico.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 506–513, 2004.
© Springer-Verlag Berlin Heidelberg 2004

software system that complies with the users´ needs if the software engineers have a precise knowledge of the way in which the organization works.

It is important to point out that the main objective of an information system is to automate certain tasks or activities in a business process, allowing the organizational actors to reach their particular goals as well as the general goals of the organization.

In a software production process that does not have the business process model as a first stage, any attempt to generate a prototype of the information system will be reduced by the incapacity to assure beforehand the real usefulness of the system in the context of the organizational tasks. However, at present, only a few research efforts focus on developing a methodological approach to eliciting and refining organizational requirements. These organizational requirements need to be used to generate the requirements model for the information system. In this way, it is possible to determine the goals that are satisfied by each use case of the requirements model. It is very important to assure that the information system integrate the functionality expected by the organizational actors for fulfill their goals and to perform their tasks.

The objective of this paper is to present a goal-based method to obtain and refine organizational requirements. To do this we propose a Goal-Refinement Tree with a new goal classification. This structure allows us to represent goals, operations, actors and dependencies between actors. We consider this information to be relevant to define a requirements model.

The paper is structured as follows: Section 2 presents an overview of the presented method. Section 3 presents the Goal-Based Organizational Modeling Method proposed in this paper. Finally, Section 4 presents the conclusions.

2 Overview of the Proposed Method

In this section, we present a general overview of the two phases that compose the proposed method.

1. Use a Goal-Based Elicitation Method to construct a Goal-Refinement Tree (GRT) that captures the organizational context.
   - Using a Goal-Refinement Strategy.
   - Using a Goal-Abstraction Strategy.
2. Use the GRT to analyze the several possibilities to satisfy goals or complete operations in the organization.

In order to illustrate our approach, we take the organization of a meeting, i.e. "*Organize a Workshop*" as a case study, and we focus on the papers review process for the Workshop. The objective of this case study is to model two different alternatives that exist when organizing such technical meetings. One of them is the "quality review process". The objective of this alternative is to select the best papers and to give quality feedback to the authors; on the other hand, we analyze the "simplified review process". The objective of this alternative is to reduce the time and effort necessary to organize the workshop. In both cases, we analyze the trade-off between the two alternatives to determine whether the goals of the organizational actors are fully satisfied.

3 The Goal-Based Organizational Modeling Method

In this section, we describe the steps shown in the previous section in detail. Section 3.1 shows the Goal-Based Elicitation Method. Section 3.2 shows the alternatives analysis to satisfy the actor's goals. The case study is used to show the application of each step of the proposal.

3.1 Step 1: Using the Goal-Based Elicitation Method

The Goal-Based Elicitation Method proposed in this paper allows us elicit the organizational goals and to represent these in a goal structure. To do this, we propose a Goal Classification, which permits us to construct a Goal-Refinement Tree using Refinement and Abstraction Strategies.

The root of the Tree represents one of the general goals of the organization. The intermediate nodes represent the groups of low-level goals for the satisfaction of a more general goal. Finally, the leaves could represent operational goals which satisfy the low-level goals or could represent goals which will be refined in succeeding modeling phases. A more detailed Goal Classification was proposed to structure the GRT.

3.1.1 Goal Classification and Goal Structure in the GRT

We propose a goal classification to structure the goals in the GRT. We present an example for each one of the goal types.

- **Operational Goal**: These goals can be satisfied by the correct state transition of one of the organizational actors[6]. There are two types of Operational Goals:
  - Operation-Dependency. In this case, the actor responsible for completing the operation depends on another actor to provide a resource or perform another operation. These kinds of Operational Goals are represented in the GRT as OP-Dep.
  - Operation Without-Dependency. In this case, the actor responsible for completing the operation does not depend on another actor to complete the operational goal. These kinds of Operational Goals are represented in the GRT as OP-WDep.
- **Achievement Goals**: These are the goals that are refined only in Operations Without-Dependency. They are represented in the GRT as *AG*.
- **Achievement-Dependency Goal**: These are goals that are refined Operational Goals, where at least one of these is an Operations-Dependency. They are represented in the GRT as *ADG*.
- **General Goals**: These are high-level goals that are used to express the business manager's point of view. Goals of this type lead directly to General Goals, Achievement Goals or Achievement-Dependency Goals.

We have defined mechanisms to structure the Goal-Refinement Tree.

- Conflict Goals: These are goals whose satisfaction leads the actors to contradictory states. They are represented by CG.
- Decomposition Goals: These represent the necessary Subgoals to satisfy a more general goal. These are represented using the link ⟶+ which links the subgoals with the goal.
- Alternative Goals: This is the case in which only one of the goals should be satisfied. These represent a decision structure to show the alternatives that exist to achieve a goal. They are represented using the link ⟶▶ which links the alternatives subgoal with the more general goal.

3.1.2 Refinement Strategy to Create the GRT

This top-down goal analysis is useful in the cases where the analyst elicits the goal of the organizational managers, who tend to express high-level goals.

In the *refinement strategy*, it is necessary to select some of the general goals of the organization and determine the subset of subgoals that permit us to satisfy it. This information is used to construct the high levels of the Goal-Refinement Tree (General Goals). It is possible to continue the refinement to detect low-level goals or operations that satisfy the high-level goals. Once the low-level goals or the operations are determined, it is necessary to find the actors responsible for accomplishing them.

Figure 1 shows a fragment of the GRT generated by the Refinement Strategy where the more general goal is "*organize a Workshop*". To achieve this goal, it is necessary to achieve the subgoals: *select papers to be presented*, *find a correct location for the Workshop, find financial aid, define a schedule,* etc. At the same time, to satisfy the goal *select paper to be presented*, it is necessary to *implement a papers review process* for the workshop.

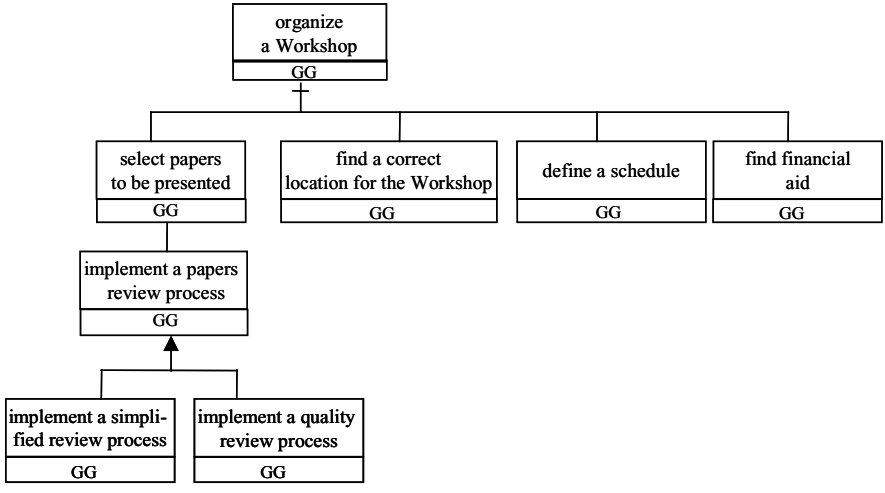


Fig. 1. A fragment of the Goal-Refinement Tree obtained with the Refined Strategy

In the refinement process, it is possible but not necessary to refine all the goals into Operation-Dependency or Operations Without-Dependency. In this way, it is possible

to leave goals to be operationalized in subsequent phases of organizational modeling. These goals are represented using a * symbol. In the case of goals that have been operationalized, it is necessary to find the actor responsible for achieving the operational goal.

3.1.3 Abstraction Strategy to Create the GRT

This bottom-up goal analysis is useful in the case where the analyst elicits the goal of the organizational actors who tend to express low-level goals.

In the *abstraction strategy*, it is necessary to detect the actors who participate in the organization. Once the organizational actors are detected, their goals and operations need to be elicited. This information is used to construct the low levels of the Goal-Refinement Tree (Operational Goals). Later, it is necessary to determine the objective of the execution of the actor operations and to determine the more general goals that are satisfied by the more specific goals of the actors.

The goals of the actor must be represented in the Goal-Refinement Tree in a direct or indirect way. The specific actor goals may need to be translated into a more general goal of the organization. For example, the goals of the Authors: *have a paper accepted in the workshop* and to *obtain feedback* could be satisfied by the goal *Send Notifications and Reviews*.

In the process of identification of the actors responsible for achieving the Operational Goal, it is possible to find dependency relationships among actors. There are dependency relationships when it is necessary for another actor to provide a resource or perform another operation to satisfy a specific operation of an actor. These dependencies must be represented in the Goal-Refinement Tree as Operation-Dependency. For example, one of the goals of the PcChair is to *Obtain a large number of quality papers*; however, to achieve this goal, the PcChair depends on the Authors to submit their papers to the workshop. In this case, it is necessary to create an Operation-Dependency called *Obtain papers*. In the Abstraction Strategy, as well as in the Refinement Strategy, it is possible but not necessary to refine all the goals into Operation-Dependency or Operations Without-Dependency. Figure 2 shows a fragment of the GRT generated by the Abstraction Strategy.

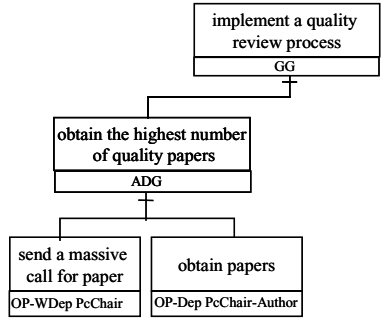


Fig. 2. A fragment of the GRT obtained with the Abstraction Strategy

3.1.4 Result of the Goal-Based Elicitation Process

As result of the Abstraction and Refinement process, we determine that it is possible to have two different alternatives to satisfy the goal *Implement a papers review process*: *implement a simplified review process* and *implement a quality review process*. In both, the goal *select paper to be presented* is achieved.

Figure 3 shows the GRT for the case *implement a simplified review process*. In this case, the strategy to satisfy the goal consists of reducing the number of reviews or of eliminating the reviews and accepting all the papers if the number of papers is limited. This last alternative was a real case of a small workshop of an important conference. In this case, due to the few papers submitted by the authors, the PcChair decided to accept all the papers without reviews and to send the acceptance notifications to the authors. Therefore, this must be analyzed as this goal model has repercussions in the satisfaction of the goals of the organizational actor.

Figure 4 shows the GRT of the case *implement a quality review process*. In this case, the strategy to satisfy the goal consists in selecting adequate PcMembers and Reviewers to review the paper. The PcChair performs the paper assignations based on the topics of interest of the PcMembers. The PcMembers are responsible for assigning the papers to the Reviewers. Finally, with the reviews, the PcChair creates a list of accepted and rejected papers and sends the notifications to the authors. As in the case of *implement a simplified review process*, it is necessary to evaluate the repercussions of this goal model on the satisfaction of the goals of the organizational actors.

3.2 Step 2: Analysing Alternatives to Satisfy Goals Using the GRT

Once the Goal-Refinement Tree is created, one of the alternatives to satisfy the organizational goal must be selected. The choice is based on trade-off analysis among the goals of the organizational actors.

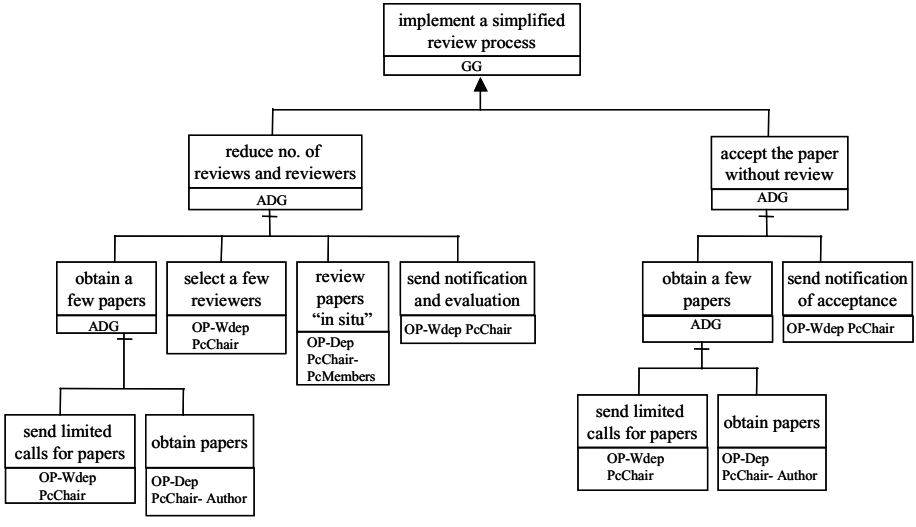


Fig. 3.The GRT of the alternative: *Implement a simplified review process*

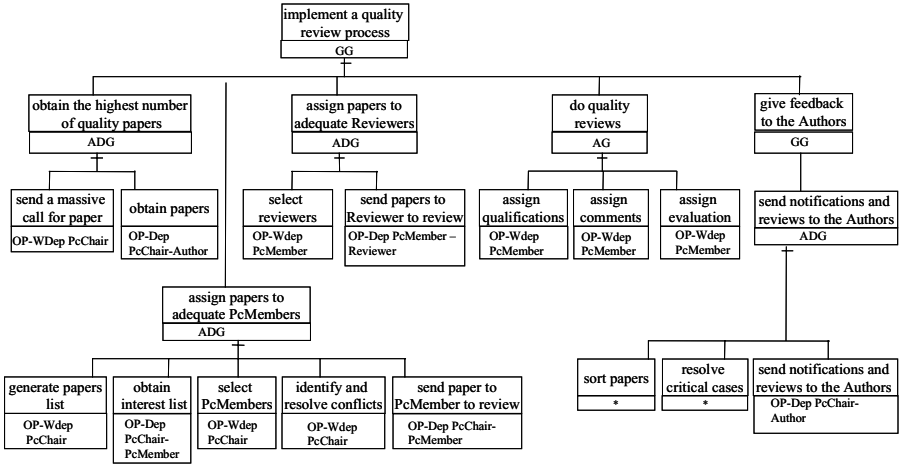


Fig. 4. The GRT of the alternative *Implement a quality review process*

The strategy at this phase is to decide the priority objectives to be satisfied in the business. For example, in the case study presented, if the goal *implement a simplified review process* is selected, the goals of the PcChair are satisfied, as there are papers to be presented and the review process is simple. However, this alternative does not satisfy the goal of the Authors. The Authors expect not only to have a paper accepted in the workshop, but also expect to obtain some (quality) feedback for their submission. Hence, the goal *implement a simplified review process* does not satisfy the Authors objectives. For this reason, we select the goal *implement a quality review process*.

The Goal-Refinement Tree can also be used to carry out obstacle analysis, conflict management and goal consolidation to generate a consistent and non-redundant goal structure. We propose using the methods and strategies proposed by KAOS [6] and GBRAM [7] to carry out these specific goal analyses.

As result of this phase, we have a Goal-Refinement Tree, which represents the objectives of the organization.

4 Conclusions

In summary, a novel goal-based organizational modeling method has been presented in this paper. The proposed method integrates a set of steps to generate a Goal-Refinement Tree that reflects the goals of each actor as well as the general goals of the organization.

The goal-refinement tree proposed in this paper allows us to decompose the high level goals in operations performed by the individual actors following a refinement strategy. In the same way, the goal-refinement tree allows us to relate each one of the tasks of the organizational actor with an organizational goal following an abstraction strategy. In this way, it is possible to generate a consistence goal structure that reflects the goal of the enterprise in a unified way.

We have applied the guidelines for organizational goal modeling to a case study showing the graphic representation of each one of the models generated.

The method proposed in this paper allows us to create goal models that can be used as a starting point for the creation of a software requirements specification, by doing this; we go a step further in the process of properly embedding early requirements engineering into the software production process.

References

1. Nunes N., J., Cuhna.J.F., WISDOM: A Software Engineering Method for Small Software Development Companies, IEEE Software 17(5), Special Issue on Software Engineering in the Small, (200), pp 113-119.
2. J. Gómez, C. Cachero, O. Pastor, On Conceptual Modeling of Device-Independent Web Applications: Towards a Web Engineering Approach, IEEE Multimedia 8(2), Special Issue on Web Engineering, (2001), 26-39.
3. Andrade Luis, Amilcar Serdanas, Banking and Management Information System Automation, proceedings of the 13th World Congress of International Federation of Automatic Control, San Francisco, USA (1996), pp. 133-138.
4. Schwabe Daniel and Gustavo Rossi, An Object Oriented Approach to Web-Based Application Design, Proceedings of the Theory and Practice of Object Systems, New York, USA (1998), pp. 207-225.
5. Oscar Pastor, Jaime Gómez, E. Infrán, V. Pelechano, The OO-Method approach for information systems modeling: from object-oriented conceptual modeling to automated prgramming, Information Systems 26(7), (2001), pp. 507-534.
6. Dardenne, A. Van Lamsweerde and S. Fickas, Goal Directed Requirements Acquisition, Science of Computer Programming, vol. 20, North Holland (1993), pp. 3-50.
7. Anton Annie, Goal Based Requirements Analysis, in Proceedings Second International Conference on Requirements Engineering. ICRE ’96, Colorado Springs, Colorado, USA (1996), pp. 136-144.

Using C++ Functors with Legacy C Libraries

Jan Broeckhove and Kurt Vanmechelen

University of Antwerp, BE-2020 Antwerp, Belgium,
{jan.broeckhove, kurt.vanmechelen}@ua.ac.be

Abstract. The use of *functors* or *function objects* in the object oriented programming paradigm has proven to be useful in the design of scientific applications that need to tie functions and their execution context. However, a recurrent problem when using functors is their interaction with the callback mechanism of legacy C libraries. We review some of the solutions to this problem and present the design of a generic *adapter* that associates a C function pointer with function objects. This makes it possible to use an object-oriented style of programming and still interface with C libraries in a straightforward manner.

1 Introduction

A functor class is a class that defines the call operator, i.e. operator()(parameter-list) as a member function [1]. As a result an object of such a class, also referred to as function object, can be used in an expression where a function evaluation is expected. Functors thus allow function call and state information to be combined into a single entity.

A common use for functors in scientific programming is the implementation of the mathematical functions [2] such as for instance the Laguerre Polynomials. The function parameters, in this case the degree of the polynomial, are passed to the function object through the constructor and do not clutter the call operator parameter list.

```
class Laguerre {
public:
    Laguerre(int degree): fDegree(degree) {}
    double operator()(double x);
private:
    int fDegree;
};
// initialize Laguerre and evaluate
Laguerre p(6);
double y = p(2.0);
```

The call to the function object is exactly similar to that of a global (C-style) function.

Another typical use for functors is structuring configurable algorithms. Consider a calculation of the derivative parametrized with the order of the difference formula and a stepsize.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 514–523, 2004.
© Springer-Verlag Berlin Heidelberg 2004

```
// declaration for the Derivative template
template<typename F>
class Derivative {
public:
    Derivative(int n, double x) : fOrder(n), fStep(x) {}
    double operator()(double x, F fctor);
private:
    int fOrder;
    double fStep;
};
// initialize the Derivative for Laguerre and evaluate
Derivative<Laguerre> der(2, 0.001);
Laguerre p(4);
double d = der(2.0, p);
```

Again these parameters are passed into the algorithm through a constructor. The advantage of structuring the algorithm this way - instead of a straight function call - is that it can be instantiated with different configurations and get passed around the program without the need to detail all parameters. This makes the program robust against future changes to the `Derivative` algorithm that add or eliminate a configuration parameter.

The `Derivative` class above shows how two software components developed independently of one another connect through the *callback mechanism*. The *caller* is an instance of `Derivative`. During its execution it calls the function whose derivative must be computed. This function is the *callee*, the Laguerre functor's call operator in our example. The callee is passed to the caller by way of the *callback function* argument in the invocation of the caller. Thus the design of the caller also fixes the type of the callee. In our example we have accomodated this by using a template construction. There are other approaches for the design of C++ callback libraries [3] [4], which support more flexible callback constructs. They are however only applicable in contexts where both caller and callee are designed in an object-oriented fashion.

We want to look at the situation that arises when the caller is part of legacy C code, some numerical library for instance. This is a common situation. Many developers are convinced of the advantages of C++ and object-oriented design, but few care to retrace their steps and reengineer their existing code to C++. The question thus arises as to how C++ functors can be hooked into C-style callbacks. When the caller is a legacy C procedure, as illustrated below, the type of the callee is necessarily that of pointer to function, determined by the function signature (the list of argument types and the return type) of the callee.

```
// declaration for the C procedure derivative
double derivative(int order, double step, double x,
                  double (*f)(double));
```

On the face of it the call operator in `Laguerre` has the appropriate signature, suggesting that its address can be used as callback function argument.

The call operator, however, is a member function and needs to be bound to a `Laguerre` object instance in order to make sense. In terms of signature there is an implicit "this" pointer in its argument list pointing to the object on which the call operator needs to be invoked. Therefore, it is not compatible with the pointer-to-function arguments accepted by a C-style callback. The possibility of binding the call operator to an object instance and then passing on this bound pointer (which would correspond to the appropriate signature) has been explicitly excluded from C++ [5].

In this contribution we want to develop a mechanism that enables object-oriented code making use of functors, to interface with legacy C libraries when the connection must be made via the C-style callback. In the following sections we develop a solution in a number of stages, each time highlighting its limits and drawbacks. We conclude with a performance analysis of our solution.

2 Solution 1: The ad hoc Wrapper

This ad hoc solution has been advanced many times, see for instance [6]. It contains some elements that also occur in our approach, so we will review it briefly. In essence one wants to be able to glue functor and C-function together, in a manner which makes it possible to deliver the glue function as an external calling interface to the functor. This glue has to be convertible to a C-style function pointer. Within a class context there is exactly one C++ construct that gives us the possibility to do this, the static member function. Static member functions are tied to the class, they are not connected with object instances. Therefore the calling convention for static member functions does not prescribe an implicit 'this' pointer argument. As a consequence, pointers to these static member functions are convertible to C-style function pointers.

The first solution involves the explicit declaration of such a static member function that will serve as a wrapper for the functor's call operator :

```
template<typename F>
class Wrapper {
public:
    typedef double (*C_FUNCTION)(double);
    Wrapper(const F& fctor) { fgAddress = &fctor; }
    C_FUNCTION adapt() { return &glue; }
private:
    static const F* fgAddress;
    static double glue(double x) { return (*fgAddress)(x); }
};

template<typename F> const F* Wrapper<F>::fgAddress = 0;
```

The `Wrapper` class holds a single static function `glue`. When a C-style function pointer is required, we supply the address of this static function. The function itself simply forwards the call to the functor using the address of the func-

tion object that has been stored in the wrapper. In order for this address to be accessible in the glue function, it is stored in a static data member.

The problem with this solution is that, when we supply a second functor of the same type, the templated wrapper class will not be reinstantiated. Thus per functor type we only have one data member (to store the functor's address), and one static function (to form the glue) at our disposal. When a second functor of the same type is adapted, one overwrites the data member containing the previous functor's address. If clients maintain the pointer to the first adapted functor, then calling that pointer after adaptation of the second functor will invoke the `operator()` on the second functor instead of the first. This situation is certainly unacceptable.

3 Solution 2: An Adapter

A second solution transfers the responsibilities for adapting functor call operators to C-style function pointers to a templated *adapter*. In order to support the adaptation of more than a single functor instance of the same type, we introduce a mapping structure. In this structure, pairs of function object addresses and associated glue functions are registered.

```
typedef double (*C_FUNCTION)(double);
C_FUNCTION adapt(FUNCTOR& f) {
  int index = indexOf(&functor);
  if (index != -1) { return fMap[index].second; }
  else {
    switch (fMap.size()) {
    case 0: fMap.push_back(make_pair(&f,&glue0)); break;
    case 1: fMap.push_back(make_pair(&f,&glue1)); break;
    case 2: fMap.push_back(make_pair(&f,&glue2)); break;
    default: throw overflow_error("Map overflow");
    }
    index = fMap.size()-1;
  }
  return fMap[index].second;
}
// glue function definition
static double glue0(double x) { return (*(fMap[0].first))(x); }
static double glue1(double x) { return (*(fMap[1].first))(x); }
static double glue2(double x) { return (*(fMap[2].first))(x); }
```

The glue functions are, as before, static member functions of the adapter class. They are coded to retrieve a function object address at a fixed position in the map, dereference the address and invoke the call operator. When a functor object is adapted for the first time, a new entry that contains the address of the functor and the address of an available static member function is added to the map. The address of that member function is returned to the client. When

`adapt` is called with a functor that has been converted before, the address of the matching member function is sought out in the map and returned. The `indexOf` operation returns the index of the slot that contains the glue function for the supplied functor object.

The adapter is templated because we do not want to resort to a typeless mapping structure. The consequent problems of casting back and forth, quickly become annoying with C++ Standard conformant compilers because they do not allow pointers to function to be cast to `void*` [1].

A disadvantage of this solution is the fact that the number of functor objects that can be adapted is hard coded into the source code. If a client requires more than the hard coded number of glue functions, the adapter's implementation needs to be adjusted. Moreover, the adapter has to support a certain maximum of adapted functor objects for functors of type F1. All subsequent instantiations of the adapter component for functors of another type F2 will need to host the same number of slots. This may result in inefficient memory usage when for example, 200 functors of type F1 and 2 functors of type F2 are adapted. Another disadvantage concerns the rigidness of the glue function's signature. Although the adapter template is parametrized on the functor's type, the signature of the glue function is still hard coded into the source code. Functors of multiple types can thus be adapted, but their call operators should all share the same signature.

A variant of this approach could use iterative macro expansion [7] to generate code. This possibility has been explored but will not be outlined here as we do not want to incur the drawbacks associated with macros.

4 Solution 3: Recursive Template Instantiation

In this section we will present a solution for generating a given number of distinct static glue functions at compile time and we develop a new adapter that is not susceptible to the advantages outlined in solution 2.

An `IndexedMap` class is used to host the mapping from adapted functors to static glue functions. The map is implemented using an STL vector that contains `pair`s of `KeyType` - `MappedType` values, because we need direct access to its elements. The `KeyType` part will hold function object addresses and the `MappedType` part glue function addresses. The map supports the same access semantics as the `adapt` function of the adapter described in section 3.

```
template <class KeyType, class MappedType, int mapCapacity>
class IndexedMap : public vector<pair<KeyType, MappedType> >
```

In order to let the compiler generate a given number of static functions, we have to upgrade the wrapper class that hosts the static glue function with an extra template parameter. If we are subsequently able to vary this parameter's value, new instantiations of the wrapper class will be made. We cannot use the functor's runtime address, because non-type template parameters are to be filled in with compile-time constants. Instead we will opt for a template `int` parameter. The idea is to supply a compile time value n for the number of static functions

that need to be generated and then perform a recursive instantiation process of the wrapper classes with 0...n as compile-time constants.

The wrapper class is configured with template parameters for the functor's type, return and argument types of the call operator, the maximum size of the `IndexedMap` and the aforementioned `int` parameter.

```
template <class CT, int mapMax, int i> class Wrapper {};

template<class Object_Type, class R, class P, int mapMax, int i>
class Wrapper<  CallOperatorTraits<Object_Type, R, TYPELIST_1(P)>,
                mapMax, i> {
public :
    typedef typename R (*FP)(P);
    typedef SingletonHolder< IndexedMap<Object_Type*,FP,mapMax>,
                             CreateStatic, NoDestroy > A2FMap;
    static typename R glue(typename P parm) {
        Object_Type* p = (A2FMap::Instance())[i].first;
        assert( p != static_cast<Object_Type*>(0));
        return (*p)(parm);
    }
};
```

All type information constituting the signature of the functor's call operator is combined by the `CallOperatorTraits` class, which implements the *traits* [8] technique. The encapsulation of type information within a traits class increases the modularity and resulting extensibility of the template structure. The operator's argument types are passed to the traits class using the `TYPELIST` construct provided by the Loki [9] library. A typelist is a container for types. It supports compile-time operations to compute the length of a type list, append to a list, etc. Loki's `SingletonHolder` class creates and holds a unique instance of the type defined by its first template parameter.

A glue function's wrapper that is instantiated with integer i, will call the `operator()` code of the functor in the i'th map entry.

For a given compile time value n we want to instantiate `Wrapper` classes with varying i parameter. We will perform these instantiations using a recursive template algorithm captured in the `GlueList` class shown below.

```
template<class CT, template<class,int,int> class Glue,
         int mapMax, int i>
class GlueList {
public:
    typedef GlueList<CT, Glue, mapMax, i-1>::typeList pList;
    typedef Glue<CT, mapMax, i> newGlue;
    typedef typename Append<pList, newGlue>::Result typeList;
};

template<class CT,template<class,int,int> class Glue, int mapMax>
```

```
class GlueList<CT, Glue, mapMax, 0> {
public:
    typedef Glue<CT, mapMax, 0> myList;
    typedef TYPELIST_1(myList) typeList;
};
```

`Glue` is the type for which we want to generate n instantiations. For the present discussion it will be the `Wrapper` class. The `GlueList` class defines a publicly available `typeList` type. At the end of the recursion, this typelist will contain all the `Glue` instantiations. In every step of the algorithm we take the list of the $i-1$'th `GlueList` and append a new instantiation of `Glue` to it. The compiler continues the recursive instantiation process until i reaches 0. At this point the specialization[10] of the `GlueList` template for $i = 0$ is instantiated and the recursion ends with a list of $n + 1$ entries.

The glue function addresses of these wrapper classes are inserted into the `IndexedMap` structure by means of a type-iterative algorithm based on recursive template instantiation (no code shown). The algorithm recurses over the typelist constructed by the `GlueList` template. In every step of the recursion, the address of the glue function belonging to the wrapper class at the head of the list is inserted into the map. Recursion continues until the tail of the typelist equals `NullType`, indicating the end of the list.

The adapter itself requires the client to supply the type of the functor objects that are to be adapted, the return and argument types of the functor's call operator and the maximum number of distinct functor objects that may be adapted. The call operator's argument types are passed to the adapter template using a typelist. Because users may want to store the pointer to the adapted function in a variable, we make the type of the returned glue function available through a public type definition. We have implemented the adapter as a singleton. A template instantiation of the adapter will yield a new adapter class, only when one of the template parameters changes in respect to previous instantiations. By using a singleton we emphasize that for a specific set of actual template parameters, only a single adapter will be allocated.

The code fragment below demonstrates the use of our final solution by adapting the Laguerre functor class defined in the introductory section. The adapted call operator is then passed on as a pointer-to-function argument of the `derivative` function contained in a C library.

```
//Define a 5-slot adapter and get the instance
typedef Adapter<Laguerre,double,TYPELIST_1(double),5> LGAdapter;
LGAdapter* ad = &LGAdapter::Instance();
//Initialize Laguerre functor and adapt it
Laguerre fctor(2);
LGAdapter::FunctionPointerType fp = ad->adapt(fctor);
//Pass it on to the derivative procedure
double res = derivative(2, 0.001, 1.0, fp);
```

5 Performance Evaluation

In this section, we compare the performance of our adapter approach with that of the commonly used ad hoc wrapper solution. The latter represents the minimal overhead. Measurements were obtained on a 2.4 GHz Pentium IV processor with 512 Kb L2 cache and 512 Mb of RAM. The adapter has been compiled and tested on the following platforms; Comeau 4.3 with a SunONE CC 5.1 backend on Solaris, gcc 3.2.2 and 3.3 on Solaris and SuSE Linux, Microsoft Visual 2003 C++ 7.1, Metrowerks C++ 8.3 and Intel C++ 7.1 on Windows XP. In this section, timings are presented for the Visual 7.1, Intel 7.1 and Metrowerks compilers.

Our C++ timing program consists of a loop which invokes a C function `libFunc` n times where n is specified as a command line parameter. This C function serves the role of a legacy C library procedure performing a callback. It accepts a pointer-to-C function, along with the current value of the loop variable. In our test setup, a functor's operator() returning the sum of its two integer arguments, will be called back through the function pointer. For the ad hoc case, this pointer will be the address of a global C function which forwards the call to the functor. For the adapted case, the pointer is obtained by adapting the functor's call operator. The library function passes an integer constant as the first parameter and the value of the loop variable as a second parameter to the functor's call operator. The code for the C function is compiled with maximum optimization level and the resulting object code is placed into a separate object file. The timing program is invoked 100 times for every value of n. This is done using a Korn Shell script executed on a Windows platform in an Interix [11] environment. The code was timed using calls to the `QueryPerformanceCounter` function resident in `Windows.h`. This function returns a 64-bit integer containing the current value of a high resolution counter. Its tick frequency is platform dependent and can be queried by calling `QueryPerformanceFrequency`. On our system, the timer generates 3579545 ticks per second which results in a timing accuracy of 279 ns. Our timing accuracy is also limited by an extra delay of 1.16 μs caused by calling `QueryPerformanceCounter` twice, before and after the loop.

Results are shown in figure 1 for the Intel 7.1 compiler with optimization level O2. The graphs show the mean time per call to `libFunc` for different values of n, augmented with error bars indicating the 95% confidence interval of the 100 samples. The graph for the adapter illustrates a 1.34 ns decline for $n = 10^3$ versus $n = 10^4$. This can be explained by the aforementioned timing accuracy since the maximal effect of the measurement error on the time per call is 1.43 ns for $n = 10^3$, declining to 0.14 ns for $n = 10^4$. The ad hoc wrapper graph shows the same decline of 1.34 ns, confirming the validity of our explanation. Timing results for different compilers and optimization levels are given in table 1 for $n = 10^7$. For all compilers and optimization levels, the variance σ^2 stays below 0.03 ns for $n = 10^7$. The results for the Visual compiler with optimization level

O2, indicate an additional overhead of 1.07 ns (10.67%) for the adapted solution in comparison to the ad hoc wrapper solution.

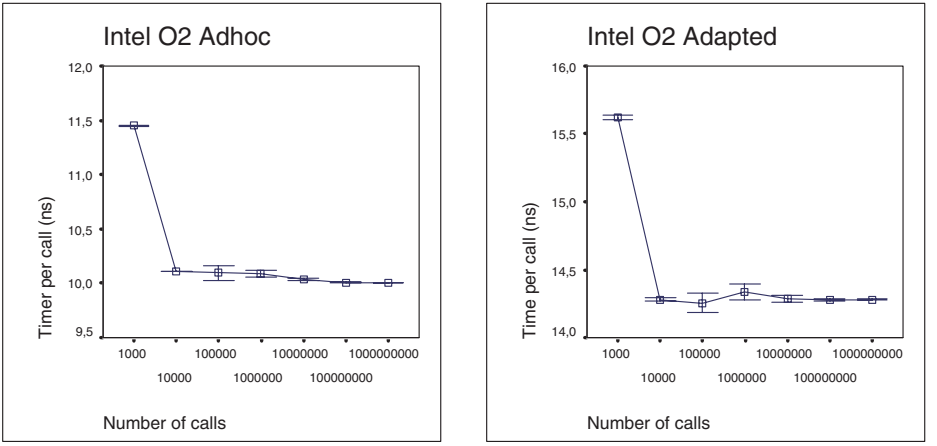


Fig. 1. Time per call to `libFunc` for the Intel 7.1 compiler on optmization level O2. The left panel shows the results for the ad hoc wrapper. The right panel shows the results for our adapter.

Table 1. Time per `libFunc` call (ns) for the ad hoc case and adapted case using different compilers and optimization levels. Results shown for $n = 10^7$.

	VC 7.1			Intel				MCW				
Optimization	O2	O1	O0	O3	O2	O1	O0	O4	O3	O2	O1	O0
Adapted	11.1	15.18	18.43	14.26	14.28	18.77	74.28	14.46	14.61	14.62	14.46	16.87
Ad hoc	10.03	10.08	10.86	10.02	10.05	10.06	17.54	11.7	11.7	11.73	11.75	13.39
Overhead	1.07	5.1	7.57	4.24	4.23	8.71	56.74	2.76	2.91	2.89	2.71	3.48

This overhead can be attributed to an extra access to the map, in order to obtain the pointer to the functor object, and an extra assertion in the `forwardCall` function to check the validity of this pointer. For all compilers and optimization levels, automatic inlining was turned on. When optimization was enabled, all compilers were able to inline the functor's call operator code into the wrapper function for both the adapted and ad hoc cases. The Intel compiler's poor performance on optimization level 0 can be partially explained by the fact that the call operator was inlined in the ad hoc wrapper function, but not in the `forwardCall` function of the `Wrapper` class. The other compilers also inlined the call operator code when optimization level 0 was used. Table 1 shows that the impact of the extra calls in the `Wrapper` class is heavily reduced by the compiler's optimizations. Overall, the Visual compiler produces the fastest code, followed

by Intel and Metrowerks. For the Intel and Metrowerks compilers the additional overhead might be deemed relatively high compared to the time taken for a call in the ad hoc case. However, in call operator implementations that involve more then a simple addition, the relative share of extra instructions introduced by our solution will be much smaller.

6 Conclusion

We have presented a solution to the problem of adapting a functor's call operator to a C-style function using recursive template programming techniques. The number of functors to be adapted has to be specified at compile time. Performance analysis has shown that the overhead of our solution relative to the ad hoc approach is small when the usual compiler optimizations are applied.

References

1. B. Stroustrup. *The C++ programming language.*. Addison-Wesley, 1997.
2. J. Barton and L. Nackman. *Scientific and engineering C++.* Addison-Wesley, 1994.
3. R. Hickey. Callbacks in C++ Using Template Functors. *C++ Report*, 7(2):42–50, February 1995.
4. P. Jakubic. Callback Implementations in C++. In *23rd Technology of Object-Oriented Languages (TOOLS23, Santa Barbara, CA, July 28-Aug. 1)*, pages 377–406. Eds. IEEE Computer Society Press., 1997.
5. M. Ellis and B. Stroustrup. *The Annotated C++ Reference* Addison-Wesly, 1990.
6. L. Haendel. The function pointer tutorials. http://www.function-pointer.org.
7. P. Mensonides. The Boost preprocessor library. http://www.boost.org/libs/.
8. N. Meyers. Traits: A New and Useful Template Technique. *C++ Report*, 7(5):32–35, June 1995.
9. A. Alexandrescu. *Modern C++ Design.* Addison-Wesly, 2001.
10. D. Vandevoorde and N. Josuttis. *C++ templates.* Pearson Education, 2003.
11. Interop Systems *Windows Services for UNIX* http://www.interix.com

Debugging of Java Programs Using HDT with Program Slicing

Hoon-Joon Kouh, Ki-Tae Kim, Sun-Moon Jo, and Weon-Hee Yoo

School of Computer Information Technology, Kyungin Women's College
#101 GyeSan Dong, GyeYang Ku, Incheon, KOREA
hjkouh@kic.ac.kr
Dept. of Computer Science and Engineering, Inha University
#253 YongHyun Dong, Nam Ku, Incheon, KOREA
{kimkitae,smpink77}@hotmail.com, whyoo@inha.ac.kr

Abstract. In the previous work, we presented HDT for debugging logical errors in Java programs. The HDT locates an erroneous function at an execution tree using an algorithmic program debugging and locates a statement with errors in the erroneous function using a step-wise program debugging. It reduced the number of programmer debugging in Java programs. But the HDT still increases the number of debugging because the size of the recent programs increases than the past programs and the number of methods is increasing. This paper proposes HDTS using a program slicing technique (PST) at the HDT. When programmers debug a Java program with HDTS, the PST can reduce the number of programmer debugging.

1 Introduction

Program debugging demands a lot of time expense and cost in a software development[1]. A debugging process is complicated for a large-scale complex program and, especially, a program written by another programmer[2]. A step-wise program debugging technique has been used in debugging programs with a variety of programming languages including Object-Oriented Programming (OOP) languages. There has been noticed a disadvantage of the step-wise program technique. That is a programmer should take a part in a debugging process and the number of debugging from programmer increases with a growing of program size. Thus, there is a hot issue that is how to reduce the number of debugging from programmer effectively[3].

In the previous work, we presented HDT[3,4] for debugging logical errors in the Java programs. The HDT locates an erroneous function at an execution tree using an algorithmic program debugging[5] and locates a statement with errors in the erroneous function using a step-wise program debugging. The HDT could reduce the number of programmer debugging in Java programs with many method calls by not debugging statements without errors[3]. But the HDT still increases the number of interactions because the size of the recent programs increases than the past programs and the number of methods is increasing.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 524–533, 2004.
© Springer-Verlag Berlin Heidelberg 2004

In this paper, we propose a Hybrid Debugging Technique with Slicing(HDTS) technique, which reduces the number of debugging form programmer by combining Hybrid Debugging Technique(HDT) and Program Slicing Technique(PST). The PST, introduced by Weiser[6], discovers correct nodes at an execution tree and correct statements in erroneous function before processing of the HDT and reduces the number of possible nodes and statements that the HDT visits. Therefore, we can reduce the number of programmer debugging at Java programs using HDTS.

The rest of this paper is organized as follows: In Section 2, we propose HDTS which combines HDT and PST. In Section 3, we extend an execution tree for HDTS. In Section 4, we walk through an example. In Section 5, we compare HDT to HDTS and conclude in Section 6.

2 HDT with Slicing in a Java Program

Hybrid Debugging Technique with Slicing (HDTS) is a technique that reduces the number of debugging from programmer by combining HDT and PST.

PST removes correct nodes at an execution tree and correct statements in erroneous functions before processing of the HDT and reduces the number of possible nodes and statements that the HDT visit. The PST can be used at the next four cases at the HDTS. First, by method that presented in [7], when a programmer answers in *incorrect* at some node of an execution tree and selects one output variable of the error, a debugger removes children nodes that have no correlation with the error(Slicing1). Second, if a programmer finds a node(method) including an error at an execution tree, a debugger removes statements that have no correlation with the error in the erroneous node. This is the concept of the most general static slicing(Slicing2). Third, when a programmer begins debugging from the following node of an execution tree again after locating and fixing a statement including an error, a debugger removes nodes that the variable related to mistake is used among nodes that do not trace yet(Slicing3). Finally, nodes that a programmer answers in *correct* and his children nodes are removed at the tree when an execution tree is built again(Slicing4). At the HDT, the first, third, fourth cases of the PST remove the unnecessary nodes at an execution tree using a static slicing method and the second case of the PST removes the unnecessary statements at a source program of the erroneous method using a dynamic slicing method. The Debugging of HDTS shows at Algorithm 1.

Algorithm 1. HDTS: HDT with Slicing

Input: Incorrect Program P
Output: Correct Program

```
1 procedure HDTS( P )
2 begin
  let  t = {(o1,c1,m1,in1,out1),..., (on,cn,mn,inn,outn) } be the
  top level trace nodes of (o0,c0,m0,in0,out0) , 1≤i≤n
3 repeat
```

```
4   Build an execution tree from P
5   Slicing4
6   n := (o0,c0,m0,in0,out0) //select a start node
7   debug := False
8   i := 1
9   if(Query((o0,c0,m0,in0,out0)) = correct) then
10     debug := True
11  else
12    if ∃t then
13      Slicing1
14      while i<n do
15        if(Query((oi,ci,mi,ini,outi)) = correct) then
    //let (ok,ck,mk,ink,outk) be the right sibling node of
    (oi,ci,mi,ini,outi)
16          if ∃(ok,ck,mk,ink,outk) then
17            i := j
18          else
19            Slicing2
20                  Statement_debug(the parent node of
(oi,ci,mi,ini,outi))
21            Slicing3
22          fi
23        else
24          i := i + 1
25        fi
26      od
27      Slicing2
28      Statement_debug((oi,ci,mi,ini,outi))
29    else
30       Slicing2
31       Statement_debug((o0,c0,m0,in0,out0))
32    fi
33  fi
34 until (debug = True)
35 write("There are no errors")
36 end
```

At Algorithm 1, HDTS traverses an execution tree from a start node to a last node and locates a lot of errors.

3 Extension of an Execution Tree

We should extend an execution tree for a dynamic slicing of nodes. The dynamic slicing of nodes uses SDG (system dependence graph)[8,9] and Linkage Grammar[10]. So this paper builds an execution tree included the dependence between input variables and output variables at each nodes, and between a node and a node. HDTS system can do the slicing of nodes using a graph reachability problem. This paper extends a set of edges, E at an execution tree (N, E, S).

$E = E_1 \cup E_2 \cup E_3$

E_1: a set of edges that represents a control dependency between a node and a node

E_2: a set of edges that represents a data dependency between a input variable and a output variable at each node. There are an internal data dependency and an external data dependency in the data dependency. An internal data dependency is a set of edges that represents a relation between a input variable and a output variable at one node. An external data dependency is a set of edges that represents the relation of the input/output variable between parent and child nodes, sibling and sibling nodes.

E_3: a set of edges that represents the relation between output variables of nodes including the same class name c and a method name m.

```
class SimpleCal  {
  int num, r1, r2, tsum;
  SimpleCal(){
      this(0);
  }
  SimpleCal(int num)  {
      this.num = num;
  }
  int decre(int num) {
      num = num - 1;
      return num;
  }
  int incre(int num) {
    num = num - 1;
    return num;
  }
 int  addsum(int  a,  int
b){
    r1 = decre(a) + b;
    r2 = incre(b) + a;
  }
```

```
int totalsum(int a, int b) {
       int r3, r4;
       addsum(a,b);
       r3 = r1 + 1;
       r4 = r2 + r1;
       tsum = r1 - num;
  }
}
public class Calculation  {
 public static void main(String args[]) {
      SimpleCal Obj1 = new SimpleCal();
      Obj1.addsum(3,4);
      SimpleCal Obj2 = new SimpleCal(5);
      Obj2.totalsum(3,4);
      System.out.println(Obj1.r1);
      System.out.println(Obj1.r2);
      System.out.println(Obj2.tsum);
    }
  }
```

Fig. 1. An example of a Java program

Fig. 2 is an extended execution tree built from a Java program of Fig. 1. And debugging system performs Slicing1 with Slicing3 using the reachability problem at the extended execution tree.

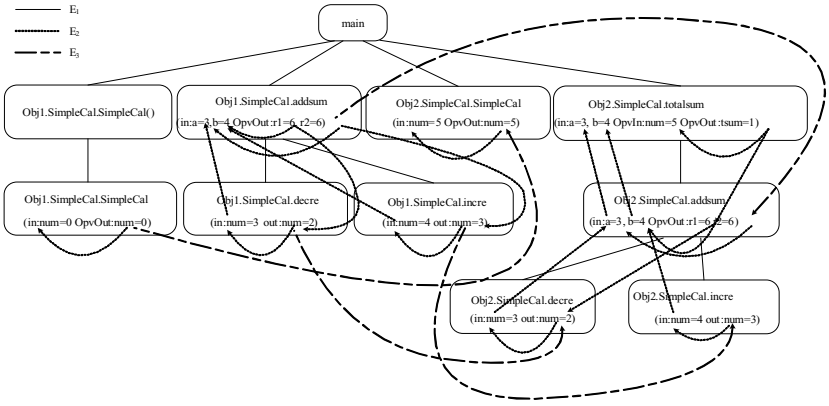


Fig. 2. An extended execution tree built from a Java program of Fig. 1

4 Debugging with HDTS

In this section, we walk through an example program of Fig. 1. We assume three cases for the debugging test from the example.

- There is an error in the method *incre*. A statement "num=num-1" is a logical error. And then a statement "num=num+1" is correct.
- There is an error in the method *totalsum*. A statement "trsum=r1-num" is a logical error. And then a statement "trsum=r1+num" is correct in the *totalsum*.

A programmer expects the result 6, 8, 11 about Obj1.r1, Obj1.r2, and Obj2.tsum, but gets the incorrect result 6, 6, 1. If a programmer starts to debug a Java program of Fig. 1, HDTS system builds an execution tree as Fig. 2. To locate logical errors in the Java program, a programmer traverses nodes of the execution tree. Debugging from the execution tree of Fig. 2 is as follows.

Step 1 : Algorithmic Debugging
Obj1.SimpleCal.SimpleCal(in:num=0 OpvOut:num=0)? $ correct
Obj1.SimpleCal.addsum(in:a=3, b=4 OpvOut:r1=5, r2=6)? $ incorrect, r2

A programmer answers in *incorrect* at *addsum* method and selects an output variable r2. The r2 is an erroneous variable.

Step 2 : Slicing1
A debugging system removes a node *Obj1.SimpleCal.decre(in:num=3 Out:num=2)* that have no corelation with a erroneous variable r2 as Fig. 3.

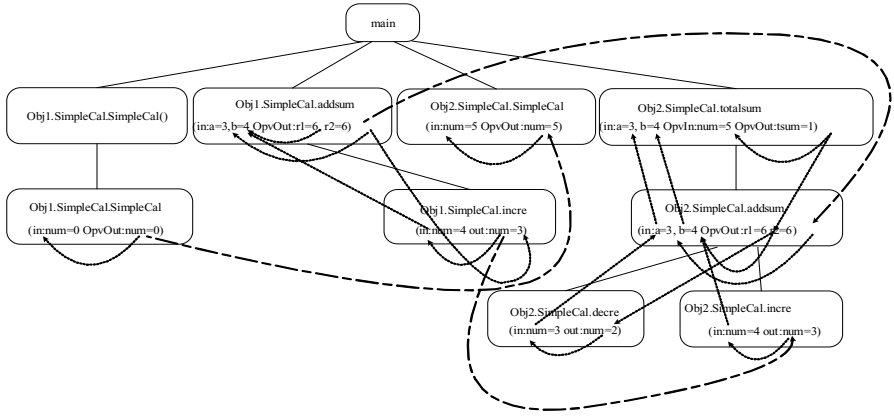


Fig. 3. An execution tree that removed child nodes dependency with an erroneous variable r2

Step 3 : Algorithmic Debugging

A debugging system asks a programmer about the next node because a node *Obj1.SimpleCal.decre(in:num=3 out:num=2)* is removed at an execution tree.

Obj1.SimpleCal.incre(in:num=4 out:num=3)? $ incorrect

The system automatically locates an erroneous node *incre* from a programmer's answer

Step 4 : Slicing2

The output variable *num* of the erroneous method *incre* is a static slicing criterion. A debugging system produces a static slicing from a slicing criterion. But there are no sliced statements in the method.

Step 5 : Step-wise Debugging

A programmer starts a step-wise debugging from a source program of method *incre* at program editor. And he traverses statements of an erroneous node using step-over, break-point, and go.

num(3)=num(4)-1 ? $ stop

A parenthesis represents the value of a variable. The programmer can modify "num=num-1" to "num=num+1".

Step 6 : Slicing3

HDTS system removes the correlative nodes with the erroneous variable *num* using the edge E_2 and the same node as a class name and a method name of the erroneous method using E_3. Fig. 4 shows the result performed Slicing3 from Fig. 3.

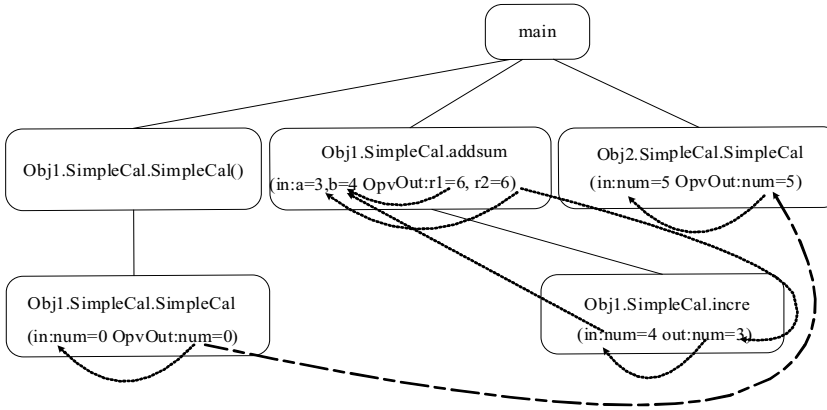


Fig. 4. A sliced execution tree from an erroneous variable *num*

Step 7 : Algorithmic Debugging

A debugging system traverses the next nodes

Obj2.SimpleCal.SimpleCal(in:num=5 OpvOut:num=5)? $ correct

HDTS system rebuilds an execution tree because a programmer traverses all nodes.

Step 8 : Slicing4

HDTS system removes the nodes that a programmer answers in *correct* at the previous tree. So, the system doesn't ask a question about two nodes when a programmer traverses a rebuilt execution tree

Obj2.SimpleCal.SimpleCal(in:num=5 OpvOut:num=5)?
Obj1.SimpleCal.SimpleCal(in:num=0 OpvOut:num=0)?

Step 9 : Algorithmic Debugging

A debugging system traverses an execution tree of Fig. 5.

Obj1.SimpleCal.addsum(in:a=3,b=4 OpvOut:r1=6,r2=8)? $ correct
Obj2.SimpleCal.totalsum(in:a=3,b=4 OpvIn:num=5 OpvOut:tsum=1)? $ incorrect
Obj2.SimpleCal.addsum(in:a=3,b=4 OpvOut:r1=6,r2=8)? $ correct

HDTS system locates an erroneous method *totalsum* from a programmer's answer.

Step 10 : Slicing2

The output variable *tsum* of the erroneous method *totalsum* is a static slicing criterion. A HDTS system produces a static slicing from a slicing criterion and removes statements "r3=r1+1" and "r4=r2+r1".

Step 11 : Step-wise program debugging

A programmer starts a step-wise debugging from the source program of a method *totalsum* at the program editor. And he traverses statements of an erroneous node.

addsum(a(3),b(4))? $ step-over
tsum(1) = r1(6) – num(5)? $ stop

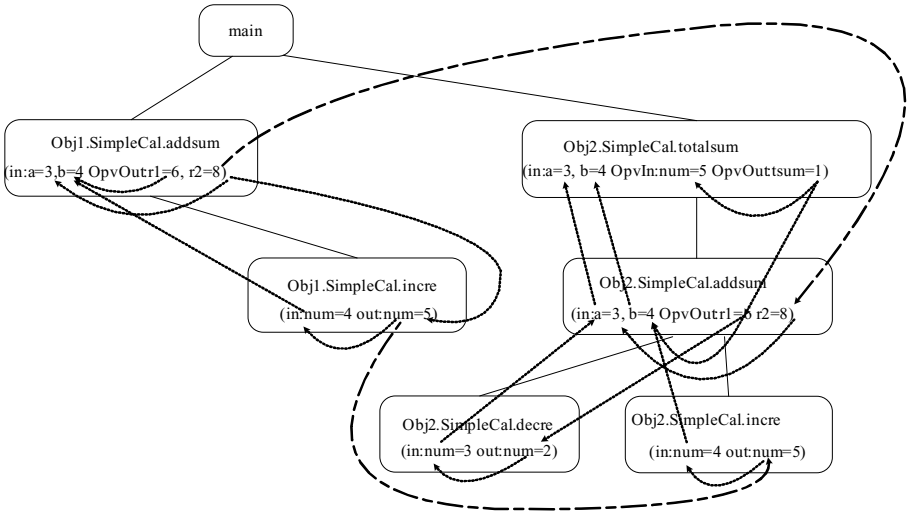


Fig. 5. A rebuilt execution tree

Then a debugger locates a statement "tsum=r1-num" including a logical error and modifies the statement to "tsum=r1+num". And the HDTS system rebuilds an execution tree because a programmer traverses all nodes(see Fig. 6).

Step 12 : Slicing4

A debugging system removes the nodes that a programmer answers in "correct" at the previous steps. The execution tree of the result is Fig. 6.

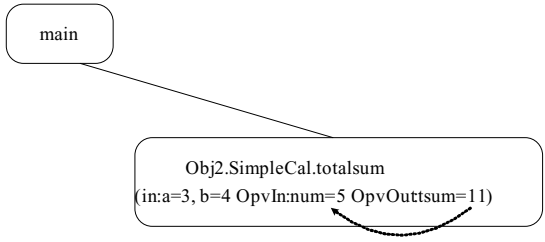


Fig. 6. An execution tree rebuilt from Fig. 5

Step 13: Algorithmic Debugging

A programmer traverses nodes from an execution tree of Fig. 6.

Obj2.SimpleCal.totalsum(in:a=3, b=4 OpvIn:num=5 OpvOut:tsum=11)? $ correct

A programmer has localized two logical errors with HDTS in a Java program.

5 Experimental Results

In this section, we present the experimental results of HDT and HDTS. In this paper, an experimental criterion is the number of interactions between a programmer and a debugging system. And we compared the proposed HDT in the previous work with the proposed HDTS in this paper. We used a Bubble Sort program and a Calculator program of Table1 for an experimental evaluation.

Table 1. Example programs for an experiment

	Methods	Logical Errors	Nodes
Bubble Sort	6	1	17
Calculator	10	2	29

We executed a Bubble Sort program to the input value '5, 2, 3, 1, 9, 8'. The number of nodes at an execution tree of the program is 17. And we executed a Calculator program to the input value '2+3*4'. The Calculator program is an arithmetic operation program using a recursive descent parser. The number of nodes at an excution tree is 29. The Bubble Sort program has a logical error and the Calculator program has two logical errors.

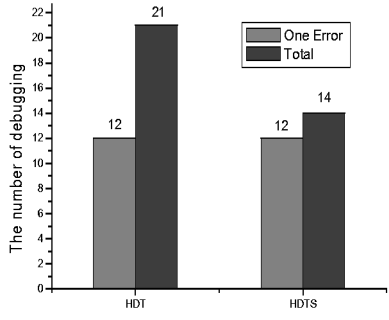


Fig. 7. The debugging of a Bubble Sort program

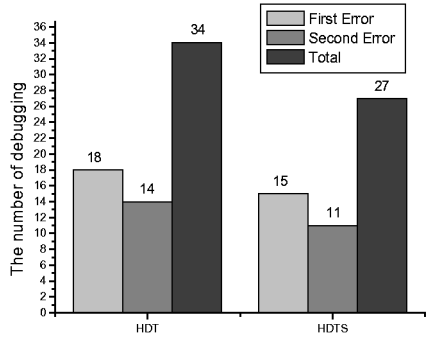


Fig. 8. The debugging of a calculator Program

Fig. 7 shows that the number of programmers' answers using HDTS is the same as HDT for locating one error but the number of total debuggings can be significantly reduced by applying Slicing4. Fig. 8 shows that the number of programmers' answers is reduced by applying Slicing1, Slicing2, Slicing3, Slicing4 for locating two errors. So the more the number of methods and statements included in methods increases, the more HDTS can reduce the number of programmers' answers than HDT. So we can say that HDTS is a suitable technique for debugging Java programs.

6 Conclusion

We proposed a Hybrid Debugging Technique with Slicing (HDTS) technique, which reduces the number of debugging from programmer by combining Hybrid Debugging Technique(HDT) and Program Slicing Technique(PST).

The PST can remove the correct nodes at an execution tree and the correct statements in the errorneous method when a programmer debugs programs using HDTS. So the PST applied to HDTS in four cases. One was a static slicing of statements at a Java source program. The others were a dynamic slicing of nodes at an execution tree. And this paper extended edges including the dependence between the input variables and the output variables at an execution tree.

Finally, this paper has walked through an example that a slicing is a suitable method at the HDTS. Our experimental results show that the proposed HDTS reduces the number of programmers' answers than HDT. Thus HDTS can help programmers to locate program errors easily.

As a next step, we will study an efficient method of locating logical errors in Java thread programs.

References

1. P. Fritzson, N. Shahmehri, M. Kamkar, T. Gyimothy, "Generalized Algorithmic Debugging and Testing," ACM LOPLAS -- *Letters of Programming Languages and Systems. Vol. 1, No. 4*, December 1992.
2. S. P. Reiss, "Trace-Based Debugging," *Proceedings of AADEBUG'93, Vol. 749 of LNCS, pp. 305-314*, Springer-Verlag, Linkoping, Sweden, May 3-5, 1993.
3. H. J. Kouh, W H Yoo: "The Efficient Debugging System for Locating Logical Errors in Java *Programs," ICCSA2003, Vol. 2667 of LNCS*, *pp.684-693*, Springer-Verlog, Montreal, Canada, May 2003.
4. H. J. Kouh, W H Yoo: "Hybrid Debugging Method: Algorithmic + Step-wise Debugging," *Proceedings of the 2002 International Conference on Software Engineering Research and Practice, pp. 342-347*, June 2002.
5. E. Shapiro, Algorithmic Program Debugging, MIT Press, May 1982.
6. M. Weiser, "Program Slicing," *IEEE Transaction on Software Engineering, Vol. Se-10, No. 4, pp 352-357*, July 1984.
7. G. Kovacs, F. Magyar, and T. Gyimothy, "Static Slicing of JAVA Programs," *Research Group on Artificial Intelligence(RGAI)*, Hungarian Academy of Sciences, Jozsef Attila University, HUNGARY, December 1996.
8. Z. Chen and B. Xu: "Slicing Object-Oriented Java Programs," *ACM SIGPLAN Notices, V.36(4):pp 33-40,* April 2001.
9. D. Liang, and M. J. Harrold, "Slicing Objects Using System Dependence Graphs," *Proceedings of ICSM '98,* pp. 358-367, November 1998.
10. S. Horwitz, T. Reps, and D. Blinkly, "Interprocedural Slicing Using Dependence Graphs," *Porceedings of the ACM SIGPLAN'88 Conference on Programming Languages Design and Implementation, SIGPLAN Notices Vol. 23(7), pp. 35-46*, Atlanta, Georgia, July, 1988.

Frameworks as Web Services

Olivia G. Fragoso Diaz, René Santaolaya Salgado, Isaac M. Vásquez Mendez, and Manuel A. Valdés Marrero

Centro Nacional de Investigación y Desarrollo Tecnológico
Interior Internado Palmira s/n Col. Palmira.
Cuernavaca, Morelos, México
{ofragoso, rene, isaacvm, valdescompany}@cenidet.edu.mx

Abstract. Object oriented frameworks represent one of the most important current approaches in component based software development for achieving high level software reuse. Frameworks for different domains others than those provided by well known development platforms are generally developed within an organization and the people who use them belong to the same organization, although their reuse scope may be widened by making them accessible to other users. One possible approach for doing this is to establish frameworks as web services. However, converting a framework to web services is possible only if the framework needs not to be extended by inheritance or aggregation, and by declaring its interfaces. This paper describes how frameworks may be established as web services and what limitations are faced when the advantages of framework based development such as customizing a framework want to be taken. An example using a framework for the statistical domain is shown.

1 Introduction

Object oriented frameworks represent one of the most important current approaches in component based software development for achieving high level software reuse. "A framework is a set of cooperating classes, some of which may be abstract, that makeup a reusable design for a specific class of software" [1]. However, the potential for reuse that a framework provides is restricted by the availability of the framework, if a framework can not be accessed; its reuse quality is almost worthless, reducing the advantages about the effort, time and expenses dedicated to its development. Frameworks for different domains others than those provided by well known development platforms, such as Java or Microsoft, are generally developed within an organization and the people who use them belong to the same organization. Their reuse potential may be increased by making them accessible to other kinds of users via web services. In this paper, an approach to establish a framework of the statistical domain as a web service is presented, and limitations and advantages are explained. A web service is an interface that describes a collection of operations that are network accessible through standardized XML messaging [2]. Although XML is the standard language that supports any web service, it is not prepared to support the extension of a framework via inheritance and aggregation relationships.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 534–542, 2004.
© Springer-Verlag Berlin Heidelberg 2004

Previous research work has been oriented towards the generation of component based web services, Yang and Papazoglou [3] proposed the realization of web services by composition taking care of the distributed flow and a tool to model the composition of services. They define abstract classes seen as library components employed as mechanisms to package, specialize, extend, reuse and classify web services versions.

Robak and Franczyk[4] described the capacity of the applications that may be obtained via composed web services. Among the capacities are user necessities, reuse, etc.

In their paper, Hailpern and Tarr [5] presented different aspects relative to web services. These authors presented the way composition of the web services may be carried out depending on their own levels, for example, where a service depends on lower level services. Participation levels may include diverse responsibilities and some times the complete functionality of the web service may not be used.

Commercial technologies such Visual Studio .NET provides characteristics for the design, specification and communication of applications with a windows based architecture and web functionality. The platform .NET [6] includes tools that assist the migration of COM components [7] to web services. While SUN [8], proposes a methodology for converting enterprise components into web services. Its conversion methodology and an example are shown at SUN Web Page with the Enterprise JavaBeans (EJB). The work proposed in this paper is closely related to the work proposed in [9] software as service SaaS, where the authors mentioned the necessity of turning software into a service by separating the possession and ownership from its use to overcome what they called "current limitations constraining software use, deployment and evolution".

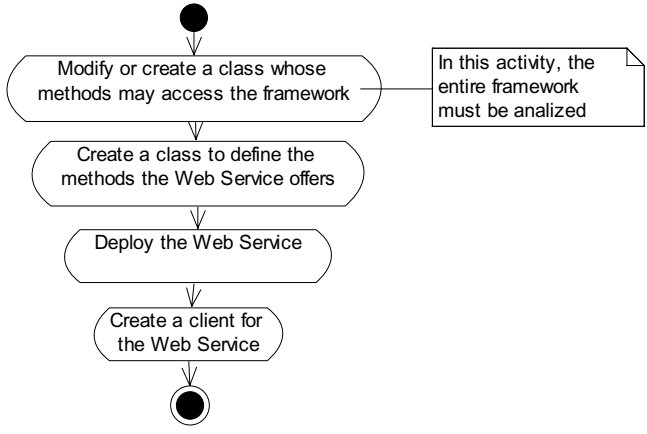


Fig. 1. General model for establishing a framework as a web service

2 General Process for Establishing a Framework as a Web Service

Figure 1 shows the model of the process for establishing a framework as a web service consisting of four activities.

The first activity consists of creating and/or modifying methods to access each one of the framework functionalities. The methods are in the class that defines the objects that access the framework.

The second activity consists of creating a new class that acts as an interface between the client and the web service.

The third activity consists of deploying the web service. The four and last activity consists of creating and/or modifying the client in order to use the web service.

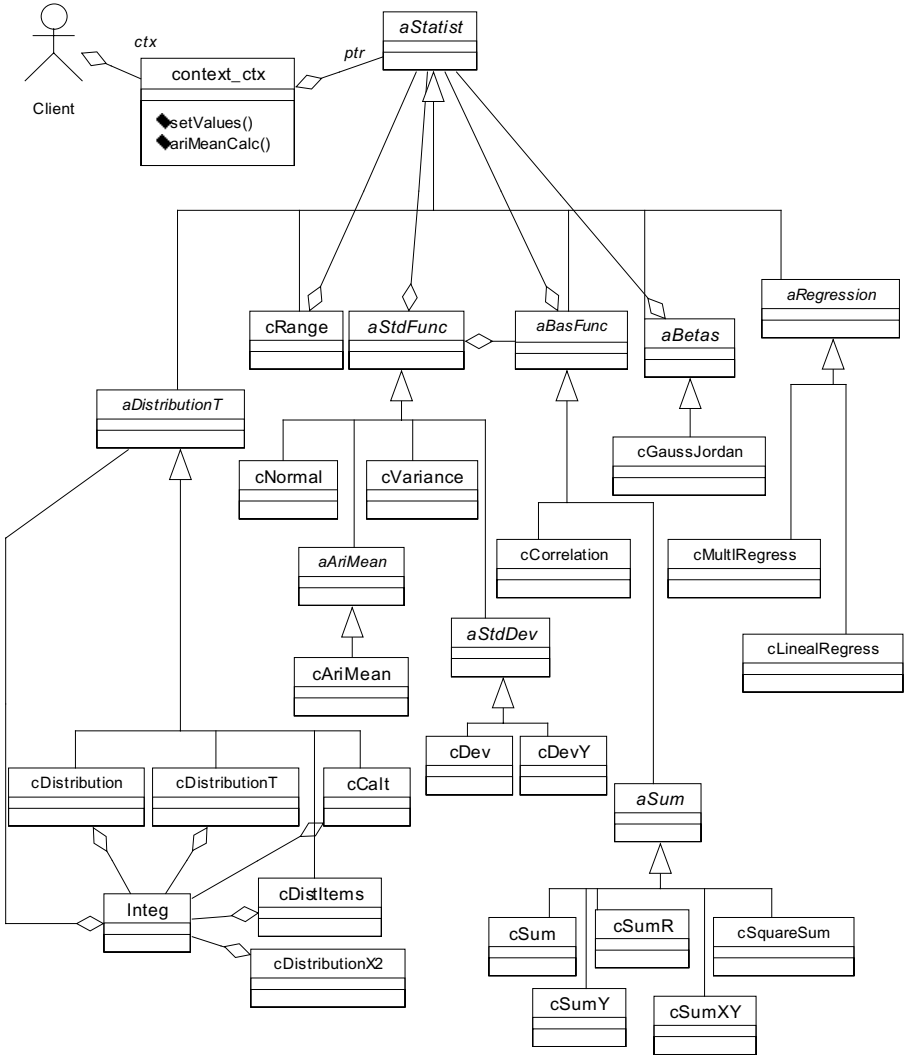


Fig. 2. Class Diagram showing the classes that belong to the Statistical Framework

3 The Statistical Framework

The object oriented framework used as an example in this paper was implemented in Java language and its structure is based on design patterns such as the Template Method, Strategy and Composite patterns [10]. In addition, some classes for sorting numbers were added. The functions of the framework refer to statistical distributions, variance and standard deviation, correlation, numerical integration, linear regression, and multiple regression. Figure 2 shows the class diagram of the statistical framework.

Figure 2 also shows the classes that implement some statistical functions. The classes names are identified with a prefixed lower case ***a*** or ***c***. ***a*** means the class is an abstract class and ***c*** means it is a concrete class.

Figure 3 shows the structure of the web service after a transformation process. The server side shows a new class named *WSStatisticInterfaces* and its relationship to an adapted *context_ctx* class. This figure also shows that the framework base class *aStatist* maintains its relationship to the adapted *context_ctx* class. The client side shows a new client class that uses the web service. The process that explains how a framework may be established as a web service is explained in section 4.

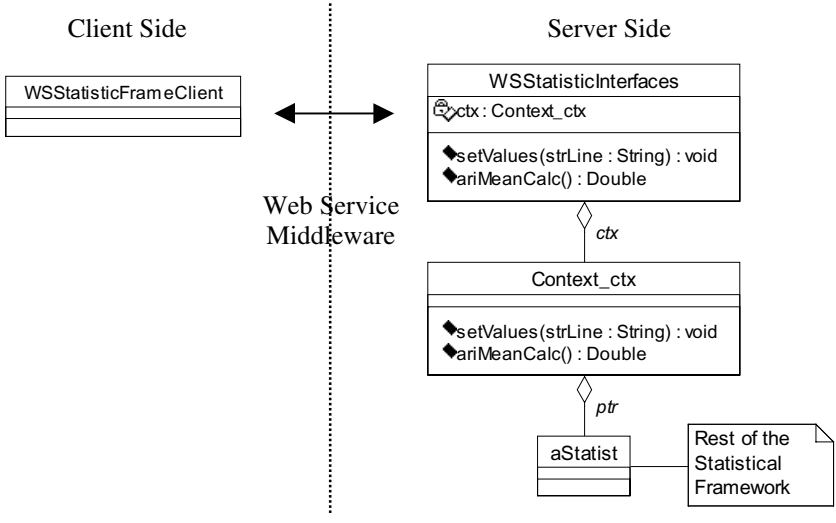


Fig. 3. Diagram showing the classes that intervene in the web service

4 The Framework as a Web Service

In this section the statistical framework is converted into web services. This has to be done by separating the client code from the framework, and adding to it the lines that instantiate the web service.

In the example below, the client code does not use a web service. The code creates an instance *x* of the *context_ctx* class, and its *setValues( )* method is called. In this

method, the values for the operation are stored in a container, and the operation of the framework that wants to be executed is called.

Example of the original code in the client class in the framework

```
import java.io.*;
public class Client
{
  public static void main (String[] args)  throws
IOException
  {
    context_ctx x = new context_ctx();
    x.setValues();
  }
}
```

The context class is a class used to transfer the messages from the client to the framework.

Example of the context class before its separation from the framework

```
Public class context_ctx
{
  public void setValues()
  {
    ...
    Code that reads from a file
    the values used by the framework
    ...
    ariMeanCalc();
  }
  public void ariMeanCalc()
  {
    aStatist ptr;
    cAriMean val= new cAriMean();
    // ptr takes cAriMean form
    ptr = val;
    ptr.InterfAlgo2(this);
  }
}
```

The code example below shows the *co*nte*xt_ctx* class after being adapted to support the web service. The methods from the framework offered by the web service, such as a*riMean*C*alc(*), were modified to return a result value. The *setValues( )* method was also modified to receive a string parameter which corresponds to a value for the methods.

New *context_ctx* class after its adaptation to support the web service

```
Public class context_ctx
{
  public void setValues(String strLine)
  {
    ...
    The values for the operation are stored in a
    container
    ...
  }
  public double ariMeanCalc()
  {
    aStatist ptr;
    cAriMean val= new cAriMean();
    // ptr takes cAriMean form
    ptr = val;
    ptr.InterfAlgo2(this);
    return dbAriMeanX;
  }
  ...
  more methods like ariMeanCalc may be included
  ...
}
```

The code in the example below corresponds to a new class named *WSStatisticInterface*s which is a class known as intermediary. The intermediary class defines the methods that the web service offers, which in this example are: *setValues(String strLinea)* and *ariMeanCalc( )*. An object of the *context_ctx* type is aggregated to the *WSStatisticInterface* class since the *context_ctx* class defines the types that an *aStatist* object may take according to the framework.

New class defining the methods the web service offers

```
package statistic;
public class WSStatisticInterfaces
{
  context_ctx ctx = new context_ctx();
  public void setValues(String strLinea)
  {
    ctx.setValues(strLinea);
  }
  public double ariMeanCalc()
  {
    return ctx.ariMeanCalc();
  }
}
```

In the example below, the code of the client uses the web service; the addresses of the WSDL document and of the web service are defined. In the line *double Mean = service.ariMeanCalc( )*, the operation required from the framework is called.

Example of a client using the web service

```
import javax.xml.namespace.QName;
import org.systinet.wasp.webservice.Registry;
import org.systinet.wasp.webservice.ServiceClient;
import iface.WSStatisticFrame;
public class WSStatisticFrameClient extends Object
{
  public static void main (String args[]) throws
Exception
  {
    WSStatisticFrame service;
    String wsdlURI =
"http://localhost:6060/WSStatisticFrame/wsdl";
    String serviceURI =
"http://localhost:6060/WSStatisticFrame/";
    // lookup service
    ServiceClient serviceClient =
ServiceClient.create(wsdlURI, WSStatisticFrame.class);
    serviceClient.setServiceURL(serviceURI);
    serviceClient.setWSDLServiceName(new
QName("urn:statistic.WSStatisticFrame",
"WSStatisticFrame"));
    serviceClient.setWSDLPortName("WSStatisticFrame");
    service = (WSStatisticFrame)
Registry.lookup(serviceClient);
    // calling methods from Web Service interface
    // Sending data....
    service.setValues("200 544");
    service.setValues("285 239");
    service.setValues("280 566");
    service.setValues("130 345");
    service.setValues("99 324");
    service.setValues("234 122");
    //Invoking any exposed methods for calc
    double Mean = service.ariMeanCalc();
  }
}
```

5 Limitations and Advantages of a Framework as a Web Service

Establishing a mature framework as a web service would be the best deal, since almost all the possible interfaces to which the framework may respond to are already defined. However, some problems arise when a framework needs to be extended in order to customize it to the client application. These problems refer mainly to the fact that when a framework is extended through an inheritance relationship, it takes the application control and instead the client code may call the framework, the framework calls the application. In a web service level it is not possible [11] since a web service is not habilitated to instantiate an object, invoke an object method, receiving the result, and releasing the object instance.

The advantages of establishing a framework as a web service from the client side view, is that a client may access to a complete set of processes of a given domain supported by the framework. From the maintenance people or developers side view, the advantages refer to the fact that adding or changing new functionalities to a framework is easier and faster, since the structure of a framework is already defined. Mechanisms such as inheritance and aggregation may be straight forward used and new interfaces may be automatically defined and be placed in the intermediary to access the framework as a web service.

6 Conclusions and Future Work

In this paper an example of how object oriented frameworks may be converted into web services has been explained. This approach is adequate for mature frameworks that do not have to be extended or that have evolved into frameworks that fully implement the functionality of the domain they represent. This limitation is due to the fact that the web services technology is very limited and it does not support the object technology, particularly the creation of class instances, messages interchange, and the receiving of a response. It works only with XML documents and the interfaces that web services offer must be defined before accessing to it.

Future work is underway to define automatically in the intermediary the interfaces the framework offers. A system for visualizing a framework documentation and user oriented is also being developed. It is a system that provides all the information from a framework such as concrete and abstract classes, interfaces, methods, and their relationships, etc, in such a way that the user can understand the functionality the framework offers and how it may be accessed.

References

1. Szyperski, C.: Component Software, Beyond Object-Oriented Programming. Addison Wesley, UK. (1999)
2. W3C . World Wide Web Consortium. www3c.org. (2003)
3. Yang, J., Papazoglou, M.: Web Component: A Substrate for Web Service Reuse and Composition. Service-Oriented Computing and Web Service Publications by Tilburg University. Tilburg, the Netherlands. (2002)

4. Robak, S., Franczyk, B.: Modeling Web Services Variability with Feature Diagrams. Proceedings of NET Object Days Conference, Germany. (2002)
5. Hailpern, B., Tarr P. L.: Software Engineering for Web Services: A Focuson Separation of Concerns. IBM Research Division Report, Yorktown Heights NY USA. (2001)
6. Fertitta, K., Sells, C.: Use ATL Server Classes to Expose Your Unmanaged C++ Code as an XML Web Service. Microsoft Corporation. MSDN Magazine. December (2002)
7. Gordon, A.: COM and COM+ Programming. Anaya Multimedia. Madrid, España. (2000)
8. Sun Microsystems. Using Exposing Model Data Via Web Services. http://java.sun.com/blueprints/webservices/using/webservbp6.html#1058898. Web Services Blueprints. (2002)
9. Turner, M.: Turning Software into a Service. In IEEE Computer.Vol. 36, No. 10. October. (2003). 38-44
10. Gamma, E.: Design Patterns. Elements of Reusable Object Oriented Software. 1st. Edition, Addison Wesley. USA. (1995)
11. Vogels, W.: Web Services are not distributed objects. In IEEE Computer Magazine. October November (2003). 59-66

Exception Rules Mining Based on Negative Association Rules

Olena Daly and David Taniar

School of Business Systems, Monash University, Clayton, Victoria 3800, Australia
{Olena.Daly, David.Taniar}@infotech.monash.edu.au

Abstract. Exception rules have been previously defined as rules with low interest and high confidence. In this paper a new approach to mine exception rules will be proposed and evaluated. Interconnection between exception and negative association rules will be considered. Based on the knowledge about negative association rules in the database, the candidate exception rules will be generated. A novel *exceptionality* measure will be proposed to evaluate the candidate exception rules. The candidate exceptions with high exceptionality will form the final set of exception rules. Algorithms for mining exception rules will be developed and evaluated.

1 Introduction

Data Mining is a process of discovering new, unexpected, valuable patterns from existing databases [2, 5]. Though data mining is the evolution of a field with a long history, the term itself was only introduced relatively recently, in the 1990s. Data mining is best described as the union of historical and recent developments in statistics, artificial intelligence, and machine learning. These techniques are then used together to study data and find previously hidden trends or patterns within.

Data mining is finding increasing acceptance in science and business areas that need to analyze large amounts of data to discover trends, in which they could not otherwise find. Different applications may require different data mining techniques. The kinds of knowledge that could be discovered from a database are categorized into association rules mining, sequential patterns mining, classification and clustering [2].

Association rule is an implication of the form X=>Y, where X and Y are database itemsets. The example could be supermarket items purchased together frequently. Two measures have been developed to evaluate association rules, which are *support* and *confidence*. Association rules with high support and confidence are referred to as *strong rules* [1, 2, 3, 4].

Negative association rule is an implication of the form X=>~Y, ~X=>Y, ~X=>~Y, where X and Y are database itemsets, ~X, ~Y are negations of database items. Negative association rules consider both presence and absence of items in the database record and mine for negative implications between database items.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 543–552, 2004.
© Springer-Verlag Berlin Heidelberg 2004

In association rules mining only the rules with high support and high confidence are considered as interesting rules. The generated patterns represent the common trends in the databases and are valuable for the marketing campaigns.

The rules with low support are just as valuable as they may contain unusual, unexpected knowledge about databases. *Exception rules* have been defined as rules with low support and high confidence [6]. A traditional example of exception rules is the rule Champagne => Caviar. The rule may not have high support but it has high confidence. The items are expensive so they are not frequent in the database, but they are always brought together so the rule has high confidence. Exception rules provide valuable knowledge about database patterns.

In this paper a new approach to mine exception rules will be proposed and evaluated. An interconnection between exception and association rules will be considered. Based on the knowledge about negative association rules in the database, the candidate exception rules will be generated. A novel *exceptionality* measure will be proposed to evaluate the candidate exception rules. The candidate exceptions with high exceptionality will be listed in the search algorithm output as exception rules.

The proposed method for mining exceptions is highly valuable in fraud detection systems, security surveillance systems, fire prevention systems etc.

Example. From a healthcare database a strong rule has been discovered that patients that have been bulk-billed can never claim the bill from the health insurance company. If some day the patients do claim the bill that has been bulk-billed (double billing), that is exception (fraud).

Bulk-billed => No Claim – Strong rule
Bulk-billed => Claim – Exception (Fraud)

2 Preliminary

In this section problem statement and related work will be discussed. Essential definitions and examples will be given in the problem statement and current research in the exception rules area will be highlighted in the related work.

2.1 Problem Statement

The search for exception rules will be based on the knowledge about strong association rules in the database. An example: we discover a strong association rule in the database, for instance shares of companies X and Y most times go up together X=>Y. Then those cases when shares of the companies X and Y do not go up together, X=>~Y or ~X=>Y, we call *exceptions* when satisfying the *exceptionality* measure explained in the next section. An algorithm for mining exception rules based on the knowledge about association rules will be proposed in the following sections.

We explain a few terms that will be used along the paper. *Itemset* is a set of database items. *Association rule* is an implication of the form X=>Y, where X and Y are database itemsets. The rule X => Y has *support s*, if *s%* of all transactions contain

both X and Y. The rule X => Y has *confidence c,* if *c%* of transactions that contain X, also contain Y.

In association rules mining user-specified minimum confidence (minconf), minimum support(minsup) are given. Association rules with support>=minsup and confidence>=minconf are referred to as *strong rules.* Itemsets that have support at least equal to minsup are called *frequent itemsets*. *Negative* itemsets are itemsets that contain both items and their negations (for example XY~Z). ~Z means negation of the item Z (absence of the item Z in the database record).

Negative association rule is an implication of the form X=>~Y, ~X=>Y, ~X=>~Y, where X and Y are database items, ~X, ~Y are negations of database items. Examples of negative association rules could be Meat=>~Fish, which implies that when customers purchase meat at the supermarket they do not buy fish at the same time, or ~Sunny=>Windy, which means no sunshine implies wind, or ~OilStock=>~PetrolStock, which says if the price the oil shares is falling, petrol shares price will be falling too.

2.2 Related Work

There have been a few research papers considering exception rules in databases [6, 7]. Existing research has presented different methods for mining interesting exception rules. In [6] the deviation analysis has been employed to distinct interesting exceptions. In [7] the information theory measures have been adopted and the interest of the exception rules has been evaluated basing on the according common sense rule. We would like to search for exception rules based on the knowledge of the negative rules. This is a conceptually new approach.

3 Proposed Exceptionality Measure and Algorithm for Mining Exception Rules in Association Mining

In this section our exceptionality measure will be explained and an algorithm will be proposed to mine exception rules based on knowledge about association rules in the database.

3.1 Exceptionality Measure

We give a few proposed definitions first. For exception rules mining instead of minsup we employ *lower* and *upper bounds,* satisfying the conditions:
0<lower bound<upper bound<minsup;
Definition 1. ***Low*** support belongs to the range [lower bound; upper bound].
Definition 2. *Infrequent itemsets* have low support.

Note that the lower bound is always greater than 0, as we are not interested in rules with 0 support or close to 0. Upper bound is lower than minsup. The lower and upper bounds are chosen specifically for each data mining application.

Definition 3. *Exception Rules* are rules with *low* support and high *exceptionality* values.

In case of exception rules in association mining the confidence measure is not applicable to evaluate the exception rules. For example, we obtain a strong rule A=>B and would like to evaluate a potential exception rule A=>Not B. The strong rule A=>B has high confidence, implying that A=>Not B cannot have high confidence. Let us say the minimum confidence is 60%. The strong rule A=>B satisfies the minimum confidence constraint, so at least 60% of database records containing A also contain B. It means that maximum 40% records containing A do not contain B. The exception rule A=>Not B has maximum 40% confidence. As confidence is not applicable for evaluating exception rules, we propose a special measure *exceptionality* to evaluate the exceptions.

Definition 4. *Exceptionality* of a candidate exception rule given the corresponding association rule is defined by the following formula:

Exceptionality(CandExc/AssocRule)=FuzzySup(CandExc)+ +FuzzyFraction(CandExc/AssocRule)+Neglect(CandExc/AssocRule)

Definition 5. Infrequent itemsets with high exceptionality are called *exceptional* itemsets.

We now explain each of the components of the exceptionality measure.

1. *FuzzySup(CandExc)*

$$\text{FuzzySup}(\text{CandExc}) = \text{FuzzySup}(\text{support}(\text{CandExc}))$$

The support of the candidate exception has to be low (see definition 1). The support of the candidate exception is not allowed to be 0 or close to 0. We define the lower and upper bound of acceptable support values. The acceptable support range we divide into regions with corresponding fuzzy support values formed by a domain expert.

For example, low support belongs to the range is 1%-5%. Then
FuzzySup(0%-0.99%)=0; FuzzySup(1%-1.99%)=1; FuzzySup(2%-2.99%)=5; FuzzySup(3%-3.99%)=2; FuzzySup(4%-5%)=1;
It is clear from the above example that the candidate rules with support in the range [2%-3%] have a better chance to gain a high exceptionality value.

2. *FuzzyFraction(CandExc/AssosRule)*

$$\text{FuzzyFraction}(\text{CandExc/AssosRule}) = \text{FuzzyFraction}\left(\frac{\text{support}(\text{CandExc})}{\text{support}(\text{AssosRule})} * 100\%\right)$$

The ratio $\frac{\text{support}(\text{CandExc})}{\text{support}(\text{AssosRule})}$ has to be relatively low. The exception rule may only be a small fraction of the corresponding association rule. Similarly to the FuzzySup(CandExc) above, the acceptable support of FuzzyFraction (CandExc/AssosRule) is divided into fuzzy regions. For example,
FuzzyFraction(0%-9.99%)=6; FuzzyFraction(10%-19.99%)=5; FuzzyFraction(20%-39.99%)=4; FuzzyFraction(40%-59.99%)=2; FuzzyFraction(60%-100%)=0

3. *Neglect(CandExc/AssosRule)*

Neglect(CandExc/AssosRule) is defined by a formula:

$$\frac{\sup(\text{only CandExc})}{\sup(\text{CandExc})} + \frac{\sup(\text{onlyAssosRule})}{\sup(\text{AssosRule})}$$

Support of *only* candidate exception is the fraction of database transactions that contain only items of candidate exception and no other items. Support of *only* association rule is defined similarly. Neglect defines what fraction of candidate exceptions and corresponding association rules occur in the database transaction on their own, while the rest of database items are absent in the transaction. The measure describes the bond between the elements of candidate exception/association rule when no other database items are present. The higher the neglect measure the stronger the bond between the items. In this case no other items could influence occurrence of the exception rule items with each other.

3.2 Classification of Exception Rules

We discuss our exception rules classification and explain the premises of mining exceptions based on the negative association rules in data bases. We suggest two general types of exception rules, which are exceptions in positive sense and exceptions in negative sense.

3.2.1 Exceptions in Negative Sense

After basic mining for positive and negative association rules in a database we obtain steady patterns of database items that occur together frequently. Let us say X and Y are database items and

$$\left.\begin{matrix} X \\ Y \end{matrix}\right\} \text{ are frequent } \quad XY \text{ is frequent} \qquad (1)$$

$X \Rightarrow Y$ high confidence

Also we obtain that

$$\left.\begin{matrix} X \sim Y \\ \text{or} \\ \sim XY \end{matrix}\right\} \text{ is infrequent} \qquad (2)$$

So we have a strong association rule (1), and we make sure that (2) are infrequent. (1) and (2) are our premises to check if one of the rules (3) has a high exceptionality, which would prove it is an exception in negative sense.

$$\left.\begin{matrix} X \Rightarrow \sim Y \\ \text{or} \\ \sim X \Rightarrow Y \end{matrix}\right\} \text{ if high exceptionality then Exception} \qquad (3)$$

Example. Consider two oil companies X and Y. Their stock normally goes up at the same time: $X \Rightarrow Y$ In the case when their shares do not go up at the same time $X \Rightarrow \sim Y$ we call the rule $X \Rightarrow \sim Y$ an exception if $X \sim Y$ is infrequent and has high exceptionality measure.

3.2.2 Exceptions in Positive Sense

After basic mining for positive and negative association in a database we obtain a steady pattern of database items. Let us say X and Y are database items and

X, Y } are frequent X~Y is frequent

X=>~Y or ~Y=>X } has high confidence (1)

Also we obtain that

XY is infrequent (2)

We have a strong negative association rule (1), and we make sure that (2) is infrequent. (1) and (2) are our premises to check if one of the rules (3) has a high exceptionality, which would prove it is an exception rule in positive sense.

X=>Y or Y=>X } if high exceptionality then Exception (3)

Example. Consider two oil companies X and Y. Their stock never goes up at the same time: X=>~Y In the case when their shares do go up at the same time X=>Y we call the rule X=>Y an exception if XY is infrequent and has high exceptionality measure.

3.3 Algorithm for Mining Exception Rules

Association rules are generated from frequent itemsets satisfying high confidence constraint. The confidence calculation is a straightforward procedure after all frequent itemsets have been generated. We do not consider the confidence calculation as it is easy and conceptually proven correct. The input of the exception rules mining algorithm are frequent 1-itemsets. The output of the algorithm is exceptional itemsets. Exceptional itemsets will become exception rules after the confidence of association rules has been checked.

We generate frequent itemsets and on each step k (k is the length of the itemset). We check the conditions (1), (2) from sections 3.2.1 and 3.2.2 and if they hold true, we check the exceptionality values for candidate exceptions. Figure 1 presents the Exceptional Itemsets Generation Algorithm.

4 Performance Evaluation

The proposed algorithm, Exceptional Itemsets Generation Algorithm, was implemented in Java 2 SDK 1.3 and tested on a PC: Celeron 1.3GHz, 128MB RAM. The test database was downloaded from the UCI Repository of machine learning databases [8]. The database used in the performance evaluation was Intrusion Detection database, which is former KDD Cup 1999 data to distinct the attacks on the network among the database records. The database represents parameters of a network over a period of time. The original database includes 40 parameters and a

```
k=1
1-freq_itemsets      // generate frequent 1-itemsets
k=2
2_candidate_itemsets          // generate candidate 2-itemsets
forEach c in 2_candidate_itemsets
 if (c frequent)       // verify the condition (1)
       if (negative_sets infrequent)        // verify the condition (2)
            {generate_2_Exc_cand_negative
            check_Exceptionality: true:   //  verify condition (3):
              if high Exceptionality of candidate
                    ExceptionalItemsets.Add }// add to the exceptional itemsets
    else
        if (negative_sets frequent)          // verify the condition (1)
            {generate_2_Exc_cand_positive
             check_Exceptionality: true:   // verify condition (3):
              if high Exceptionality of candidate
                   ExceptionalItemsets.Add } // add to the exceptional itemsets
 k++
```

Fig. 1. Exceptional Itemsets Generation Algorithm

vast number of records. Most of the parameters are continuous so in this work the simplified model of 10 parameters and 10 thousand attributes was employed. The parameters are listed in Figure 2. The parameters are either continuous or discrete.

The database sample has been chosen randomly from the original database. The continuous parameters values have been divided into ranges according to min/max values.

Exception rules mining starts with 1-frequent itemsets mining. The 2-candidate are then derived from 1-frequent itemsets and the minsup of 2-candidate itemsets is evaluated. The negative subsets are generated from 2-candidate itemset. If minimum support of 2-candidate itemset is greater or equal minsup and support of one of negative subsets is less than minsup, the pair is the candidate exception (and vice versa). For instance, for a candidate X Y Z, negative subsets {~X Y Z, X~Y Z,X Y~Z, ~X~Y Z,~X Y~Z,X~Y~Z} are generated. To verify the support of negative subsets, a special formula has been developed and tested:

Sup(negativeSubset)=

$$\mathrm{Sup}(\mathrm{positiveItems}) + \sum_{i=1}^{\mathrm{negNumber}} (-1)^i * \mathrm{Sup}(\mathrm{genSubset}(\mathrm{negativeItems}, i) + (-1)^i * \mathrm{Sup}(\mathrm{allItems})$$

In the negative subset V~X Y~Z positive items are V Y, negative items are X Z, all items are V X Y Z. genSubset generates all possible 1, 2, i, … combinations of negative items and concatenates them with positive items. Negative number is number of items negations in the itemset.

Sup(V~X Y~Z)=Sup(VZ)-Sup(VXY)-Sup(VYZ)+Sup(VXYZ);

negNumber(V~X Y~Z)=2;

There is no need in additional database scans to calculate the support of negative subsets. The support is calculated based on items positive support updated while searching frequent itemsets.

1. duration	length (number of seconds) of the connection	cont
2. protocol type	type of the protocol, e.g. tcp, udp, etc	discrete
3. flag	normal or error status of the connection	discrete
4. src_bytes	number of data bytes from source to destination	cont
5. dst_bytes	number of data bytes from destination to source	cont
6.urgent	number of urgent packets	cont
7.hot	number of ``hot'' indicators	cont
8.logged_in	1 if successfully logged in; 0 otherwise	discrete
9.Num_compromised	number of ``compromised'' conditions	cont
10.Num_file_creation	number of file creation operations	cont

Fig. 2. Network parameters

Figure 3 presents a graph of dependency between minimum support value and number of generated exception rules. Positive exceptions mean exceptions in positive sense, negative exceptions mean exceptions in negative sense (see 3.2.1, 3.2.2). Exception rules pictured in Figure 3 are the rules with the highest exceptionality measure among the candidate exception rules. The exceptions rules number is the average value for all minsup values.

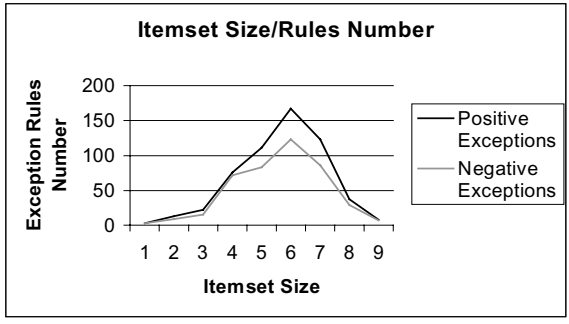


Fig. 3. Itemset Size/Rules Number Graph

Figure 4 presents minimum support value /number of generated exception rules dependency. The graph changes direction of falling/rising at different minsup values.

In Figure 5 there are a few samples of generated exception rules featuring high exceptionality value. Our algorithm generates exceptional itemsets that become exception rules after computationally simple confidence value verification. Frequent itemsets represent strong rules in the database. When an exception based on strong rule has been generated, it indicates something unusual like invasion detection in the network.

Fig. 4. Minsup/Execution Time Graph

Positive Exceptions	**Frequent Itemset**: protocol_type=tcp flag=SF urgent=0 dst_bytes C[0;500 000b] logged_in=1 **Exceptional Itemset:** protocol type=tcp flag=Not SF urgent=0 dst_bytes C[0;500 000b] logged_in=0
	Frequent Itemset: urgent=0 dst_bytes C[0;500 000b] hot C [0,9] logged_in=1 num_compromised=0 num_file_creations=0 **Exceptional Itemset:** urgent=0 dst_bytes C[0;500 000b] hot C [0,9] logged_in=1 num_compromised>0 num_file_creations>0
Negative Exceptions	**Frequent Itemset:** flag=SF source_bytes C [0;10000b] num_file_creations=0 **Exceptional Itemset:** flag=SF source_bytes C [0;10000b] num_file_creations>0
	Frequent Itemset: flag=REJ logged_in=0; **Exceptional Itemset:** flag=REJ logged_in=1;

Fig. 5. Samples of generated exceptions

5 Conclusion and Future Work

The project considers the interconnection between negative association rules and exceptions rules. The exceptions rules mining algorithm employs the knowledge about the negative association rules in the database and generates candidate exceptional itemsets. The candidate itemsets exceptionality measure is then verified. If the exceptionality satisfies the minimum exceptionality constraint, the candidate exceptional itemsets will be listed in the output of the algorithm as exceptional itemsets.

In the future work we are going to consider temporal exceptions, which are the temporal patterns in the database related with negative association rules and changing over time. Additional measures will be considered to distinct the temporal exceptions in the database.

References

[1] Agrawal, R., Imielinski, T. and Swami, A. "Mining association rules between sets of items in large databases". *Proceedings ACM-SIGMOD Int. Conf. Management of Data*, pp. 207-216, Washington, D.C., May 1993

[2] Chen, M., Han, J. and Yu, P. "Data Mining: An Overview from a Database Perspective" *IEEE Transactions on Knowledge and Data Engineering*, vol. 8, no. 6, pp. 866-883 1996

[3] Agrawal, R., Mannila, H., Srikant, R., Toivonen, H. and Verkamo, A. "Fast discovery of association rules" In *Fayyad et al, Advances in Knowledge Discovery and Data Mining, AAAI Press* pp. 307-328, 1996

[4] Agrawal, R. and Srikant, R. "Fast algorithms for mining association rules in large databases" *Proceedings of the 20th International Conference on Very Large Data Bases*, pp. 487-499, 1994

[5] Savasere, A., Omiecinski, E. and Navathe, S. "An efficient algorithm for mining association rules in large databases" *Proceedings of the International Conference on Very Large Data Bases,* pp. 432-444, 1995

[6] Liu, H., Lu, H., Feng, L. and Hussain, F. "Efficient Search of Reliable Exceptions" *Proceedings of the 3rd Pacific-Asia Conference on Knowledge Discovery and Data Mining* pp. 194-203, 1999

[7] Hussain, F., Liu, H., Suzuki, E. and Lu, H. "Exception Rule Mining with a Relative Interestingness Measure" *Proceedings of the 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining* pp. 86-97, 2000

[8] Blake, C.L. & Merz, C.J. UCI Repository of machine learning databases [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, CA: University of California, Department of Information and Computer Science, 1998.

A Reduced Codification for the Logical Representation of Job Shop Scheduling Problems

Juan Frausto-Solis[1] and Marco Antonio Cruz-Chavez[2]

[1] Department of Computer Science, ITESM, Campus Cuernavaca
Paseo de la Reforma 182-A, 62589, Temixco, Morelos, MÉXICO
juan.frausto@itesm.mx
[2] Faculty of Chemical Sciences and Engineering, Autonomous University of Morelos State
Av. Universidad 1001, Col. Chamilpa, 62270, Cuernavaca, Morelos, MÉXICO
mcruz@buzon.uaem.mx

Abstract. This paper presents the Job Shop Scheduling Problem (JSSP) represented as the well known Satisfiabilty Problem (SAT). Even though the representation of JSSP in SAT is not a new issue, presented here is a new codification that needs fewer clauses in the SAT formula for JSSP instances than those used in previous works. The codification proposed, which has been named the Reduced Sat Formula (RSF), uses the value of the latest starting time of each operation in a JSSP instance to evaluate RSF. The latest starting time is obtained using a procedure that finds the critical path in a graph. This work presents experimental results and analytical arguments showing that the new representation improves efficiency in finding a starting solution for JSSP instances.

Keywords: Job shop scheduling, the propositional satisfiability problem (SAT), Latest starting time, SAT formula.

1 Introduction

The Job Shop Scheduling Problem (JSSP) is one of the most relevant problems in manufacturing processes because the efficient resource management is a critical requirement. The JSSP is considered a very difficult problem, and, in computer sciences, is cataloged as an NP-hard optimization problem [1]. This indicates that there is not a deterministic algorithm to solve the problem, however, at the present time algorithms have been designed to solve certain instances of JSSP. Various approaches have been proposed for solving the JSSP using several models. Two of the most commonly used models are disjunctive graphs [2] and constraints satisfaction [3]. These models can be classified as search methods and optimization methods. The search methods can very quickly find a feasible solution to a JSSP instance, but unfortunately there are no guarantees that the solution found is the optimal one. However, search methods can provide a starting point. The methods based on satisfiability [4], [5], and priority rules [6] are some examples of search methods. Shifting Bottleneck ([7], [8]) is a special type of search method that has a better performance that most others, but only in small instances. On the other hand, optimization methods attempt to find the best solution to a JSSP instance or at least one that is closer to a

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 553–562, 2004.
© Springer-Verlag Berlin Heidelberg 2004

global optimum. Branch and Bound [9], Simulated Annealing [10], and Genetic Algorithms [11] are among the principal optimization methods. Even though JSSP optimization methods are outside the scope of this paper, it is important to mention that most of these methods require a starting feasible solution at the beginning of their process [12]. It is advantageous for all of these methods to find this starting solution as fast as possible. A very attractive possibility to the challenge of quickly finding a feasible solution is mapping a JSSP instance as a SAT problem [13] in such a way that the solution of the SAT problem is a feasible solution of the JSSP instance. Even though the codification of the problem of scheduling SAT is not a new issue [4], it is important to find alternative ways to codify JSSP as SAT for many reasons. First of all, because JSSP is NP hard, new SAT codification of JSSP is important in and of itself. Another reason is that for certain kinds of problems, a particular SAT codification can provide a feasible solution very quickly [5].

In this paper a SAT codification for JSSP is presented which is based on a previous one proposed by Crawford and Baker [5]. This codification is a reduced SAT formula in which the solution obtained is a feasible starting solution (a feasible schedule) of a JSSP instance, which can then be used in many JSSP optimization methods [12].

In order to get the satisfiability of the reduced logic formula one could use the well-known SAT solvers, such as GSAT [14], WalkSat [15], TABLEAU [16] and others. One could also think about using a solver in a testing plan in order to probe the efficiency of the two codifications. In this case, rather than proceed with a testing approach of the new codification, analytical arguments are presented, showing that the new Reduced Sat Formula has a smaller number of clauses, resulting in a more efficient performance than the methods which have been proposed previously.

2 Background

In order to develop the reduced codification, the following concepts were used: the JSSP, representation of the JSSP as a disjunctive graph, and SAT codification of JSSP. These concepts are explained here to give the reader background in order to be able to understand the reduced codification.

2.1 The Job Shop Scheduling Formulation

A JSSP consists of a set of jobs $J=\{j_1, j_2, \ldots, j_n\}$, a set of machines $M=\{m_1, m_2, \ldots, m_m\}$ and a set of operations $O=\{1, 2, 3, \ldots\}$. Each operation i is defined by six elements: (1) a machine m_j in which it will be processed, (2) a job j_k to which it belongs, (3) a time of processing p_i, (4) a ready time r_i, (5) a starting time s_i and (6) a deadline d_i. The ready time indicates the earliest time at which the operation can start and the deadline is the time by which the operation must be completed.

The operations of a JSSP have the following relations: For each pair (i, j) of operations that belong to the same job, a precedence relation exists. In addition, any single machine cannot execute simultaneously more than one operation.

These elements and relations allow the formulation of the JSSP as a constraint satisfaction problem (CSP) [3] that is known as Job Shop Deadline Scheduling. This for-

mulation is shown in Table 1, where the subscripts *i* and *j* are associated with two distinct operations of the problem.

Table 1. The constraints of a JSSP as a constraint satisfaction problem

Constraint	Interpretation
$s_i \geq 0$	*Starting time constraint:* The starting time of the operation *i* must be non-negative.
$s_i + p_i \leq s_j$	*Precedence constraint:* The operation *i* must be complete before *j* can begin.
$s_i + p_i \leq s_j \vee s_j + p_j \leq s_i$	*Resource capacity constraint*: The operations *i* and *j* are in conflict. They require the same resource and they cannot be scheduled concurrently.
$r_i \leq s_i$	*Ready time constraint*: The operation *i* cannot begin before its ready time.
$s_i + p_i \leq d_i$	*Deadline constraint:* The operation *i* cannot finish after its deadline.

2.2 Disjunctive Graph of the JSSP

The JSSP can be represented using a disjunctive graph [2], [17]. This graph is a 3-tuple G = (N, A, E) where N is a set of nodes representing the operations of the problem. A and E are sets of arcs that symbolize precedence constraints and resource capacity constraints respectively. Precedence constraints are represented by conjunctive arcs, whereas resource constraints are represented by disjunctive arcs.

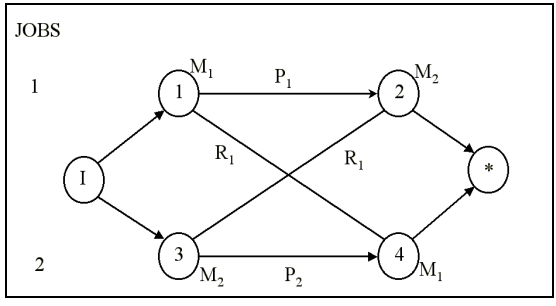


Fig. 1. Representation of a two job and two machine JSSP using a disjunctive graph

The operations connected by conjunctive arcs are those that belong to the same job, and the operations connected by disjunctive arcs are those which are executed by the same machine. Figure 1 represents a disjunctive graph where the operations 1 and 2 belong to job 1 and the operations 3 and 4 belong to job 2. In this figure, the precedence constraints are represented by P1 and P2, and the resource capacity constraints are represented by R1 and R2.

2.3 SAT Codification of JSSP

The objective of the SAT problem is to confirm or deny the existence of an assignment of truth-values for the literals of a logic formula which make the formula true.

A SAT formula is usually written in its conjunctive normal form (CNF) that has the following three features: (1) a conjunction F of clauses F_i, i.e. $F = F1 \wedge F2,\ldots, \wedge Fn$, (2) each clause F_i is a disjunction of literals $X_i v \ldots . v X_k$, (3) each literal X_j is a Boolean variable (negated or not). Crawford and Baker [5] propose a SAT codification for the JSSP based on the formulation of the JSSP as one CSP. This SAT codification is shown in Tables 2 and 3. In these tables, the subscripts i and j are associated with two distinct operations. The subscripts r_i, d_i and t are times, r_i representing the ready time and d_i representing the deadline. In this SAT codification, a JSSP instance is the set of clauses F, each CNF in Table 2 and 3 being a clause of F. In this way a JSSP instance is codified, or represented, by a SAT problem F, in which the solution is feasible for the JSSP instance.

Table 2. Logical clauses for the JSSP represented as a CSP

Constraint	CNF
$s_i + p_i \leq s_j$	$pr_{i,j}$
$s_i + p_i \leq s_j \vee s_j + p_j \leq s_i$	$pr_{i,j} \vee pr_{j,i}$
$s_i \geq r_i$	$sa_{i,ri}$
$s_i + p_i \leq d_i$	$eb_{i,di}$

Table 3. The coherence conditions for the SAT codification of a JSSP

Coherence condition	CNF	Interpretation
$sa_{i,t} \rightarrow sa_{i,t-1}$	$\sim sa_{i,t} \vee sa_{i,t-1}$	*Coherence of sa:* If i starts at or after time t, then it starts at or after the time $t-1$.
$eb_{i,t} \rightarrow eb_{i,t+1}$	$\sim eb_{i,t} \vee eb_{i,t+1}$	Coherence of *eb*: If i ends by time t, then it ends by time $t+1$.
$sa_{i,t} \rightarrow \sim eb_{i,t+pi-1}$	$\sim sa_{i,t} \vee \sim eb_{i,t+pi-1}$	*Coherence of* p_i*:* If i starts at or after time t, then it cannot end before time $t+p_i$.
$sa_{i,t} \wedge pr_{i,j} \rightarrow sa_{j,t+pi}$	$\sim sa_{i,t} \vee \sim pr_{i,j} \vee sa_{j,t+pi}$	*Coherence of* $pr_{i,i}$*:* If i starts at or after time t, and j follows i, then j cannot start until i is finished.

3 Reduced SAT Codification

The codification presented in the last section is complete because it represents all the constraints of the Job Shop Deadline Scheduling Problem with logical formulas; however for this codification, one should build and then evaluate many clauses. It would be advantageous to have a reduced codification of any JSSP instance, where the number of clauses is smaller, so in principle, the efficiency to build and evaluate the respective SAT formula is increased. The reduced codification of the JSSP is based on two concepts, the reduction of clauses and the determination of the latest starting time (LST). These two fundamental concepts are explained in the following sections.

3.1 Reduction of Clauses

It is possible to significantly reduce the number of clauses that compose the complete SAT codification, according to the following analysis.

First, all of the clauses in Table 2 can be eliminated for any JSSP instance because the respective clauses will automatically be true. They are true because any schedule proposed has a known sequence in which every pair of operations has a known order**.** In a schedule, the LST (latest starting time) of an operation is the latest time in which the operation can start. When the LST is calculated, the ready time is assigned to this time (r_i = LST) and the deadline can be determined ($d_i = r_i + p_i$). Table 4 describes the form in which the constraints can be eliminated. Given this treatment, all the clauses in Table 2 are true which means they can be eliminated.

Table 4. Conditions for eliminating the scheduling constraints in a reduced codification

Constraints which can be eliminated	Rationale
Precedence constraints $(pr_{i,j})$	For any JSSP, the precedence constraints are considered to define the problem and the equivalent clause $pr_{i\,i}$ is always true. These clauses can be eliminated from the codification because their truth-values are always true.
Resource capacity constraints $(pr_{i,j} \vee pr_{j,i})$	Because the resource capacity constraints specify the use of the same machine by two operations, and because the schedule for the problem is defined, the resource capacity constraints can be exchanged for the precedence constraints in the logical representation. When this is done, it is possible to see that these clauses can be eliminated from the codification because their truth-values are always true.
Ready time constraints $(sa_{i,ri})$ and deadline constraints $(eb_{i,di})$	If the LST of each operation in the defined schedule is found, the ready times and the deadlines of each operation can be determined. If these times are known, the clauses in the codification that contain $sa_{i\,ri}$ or $eb_{i,di}$ will be always true, so they can be eliminated from the codification.

For the clauses that represent the coherence conditions (Table 3), if t is the LST of the operation ($t=r_i$), then the literals $sa_{i,t-1}$, $\sim eb_{i,t}$ and $\sim eb_{i,t+pi-1}$ are all true. The clauses with these literals can be eliminated from the codification and the clauses that remain are the clauses that are used to evaluate the coherence of pr_{ij}. Table 5 presents the justification for realizing this elimination process.

The key of this reduction is the determination of the LST for each operation in the defined schedule.

It should be noted that RSF is formed only by the coherences pr_{ij} presented in Table 3. A SAT solver like GSAT [14], WalkSat [15], TABLEAU [16] and many others can be used in order to find a solution to the SAT problem of RSF which represents a feasible schedule. These solvers assign truth-values to the variables pr_{ij} in the RSF. Next, the LST (latest starting time) for each operation (i, j) is calculated. With the LST and truth-values of the variables involved in RSF obtained, the solver begins to prove the satisfiability of RSF.

Because all the eliminated clauses are always true, it is not useful to maintain them in the complete formula when a SAT solver is used. While the RSF presented here has fewer evaluations than the complete SAT codification, it requires that the LST be

found. One can observe that the complete SAT codification requires the calculation of the *t* times using a procedure not described by Crawford and Baker [5].

Table 5. Justification for eliminating the coherence conditions in a reduced codification

Constraints which can be eliminated	Rationale
Coherence of *sa*	Because *t* is the LST, then $t=r_i$. For clauses with the form $\sim sa_{i,t} \vee sa_{i,t-1}$, the literal $\sim sa_{i,t}$ is false, and $sa_{i,ri}$ is true. The literal $sa_{i,t-1}$ is true because the operation *i* starts at r_i which is equal to *t* and s_i is after r_{i-1}. Therefore, the clauses are always true and they can be eliminated.
Coherence of *eb*	Because t is the LST, then $t=r_i$. For clauses with the form $\sim eb_{i\,t} \vee eb_{i\,t+1}$, the literal $\sim eb_{i\,t}$ is true and $eb_{i\,ri}$ is false. Because the operation *i* cannot finish at its ready time or before, the literal $eb_{i\,t+1}$ is false and $eb_{i\,ri+1}$ is also false. Therefore the clauses are always true and they can be eliminated.
Coherence of p_i	Because *t* is the LST, $t=r_i$. For clauses in the form $\sim sa_{i\,t} \vee \sim eb_{i\,t+ni-1}$, the literal $\sim sa_{i\,t}$ is false and $sa_{i\,ri}$ is true. The literal $\sim eb_{i\,t+ni-1}$ is true and $eb_{i\,ri+ni-1}$ is false because the operation *i* cannot finish before of r_i+p_i. Therefore the clauses are always true and they can be eliminated.

3.2 Latest Starting Time

The LST of a given operation is equal to the critical (longest) path generated in a digraph between the given operation and the initial operation (*I* in Figure 1). The value of the critical path of an operation *i* is the sum of the processing times of all the operations in the sequence between *I* and *i*. The LST is used as the start time s_i of the operation *i*.

Figure 2 shows one example of two operations (*i* and *j*) processed by the same machine. In this figure, the ready time of the operation *j* (r_j) and the deadline of the operation *i* (d_i) are shown when the resource capacity constraint is transformed into one precedence constraint. These times are valid if the resource capacity constraint between *i* and *j* is not violated.

In this example, the LST of the operations *i* and *j* can be calculated (t_i and t_j, respectively), so the finish time of the operation *i* is $t_i + p_i$. If t_j is greater than or equal to $t_i + p_i$, the clause $\sim sa_{i,t} \vee \sim pr_{i,j} \vee sa_{j,t+pi}$ is true and the ready times will always be valid for each operation. If t_j and $t_i + p_i$ are equal to each other, r_j represents the finish time of the operation *i* ($r_j = t_i + p_i$) and is valid because the operation *j* will start at the time t_j (the operation *j* starts when the operation *i* finishes). The deadlines for these operations are valid, as is shown for operation *i* in Figure 2. In this example, the operation could slow its finish (d_i) until the beginning of the operation *j* (t_j). In this case, (e.g., $d_i < r_j$), there would be idle time, a few moments when the machine would not be working, between the end of one operation and the beginning of another.

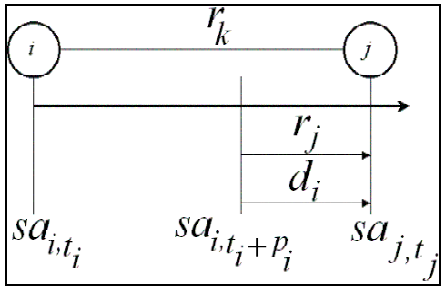


Fig. 2. Graphic representation of the resource capacity constraint between two operations (with their LST's) that have a defined sequence

3.3 Obtaining the Latest Starting Time

The general problem in obtaining the longest path in a graph is classified as NP-complete [18]. A certain relaxation is required to find the path in an efficient form. In this work, a method based on the approach proposed by Adams, et al. in [7] is used. In this method, Hamilton routes from each of the machines are taken for a possible schedule from the graph that represents the JSSP. This simplification of the directed graph generates a binary search tree for the schedule. The problem becomes finding the longest path between two nodes of the graph which can be solved in polynomial time [7], [18], with a complexity of *O(N)*, where *N* is the number of nodes generated in the binary search tree. In the search tree, the root of the tree is the operation that is needed to obtain the LST, and contains each possible route to arrive at the initial node (operation I). Each node of the resulting tree has a maximum of one successor and one predecessor.

This search tree is used to determine the LST of all the operations of the possible schedule. The LST of each operation is the critical path from this operation to the initial node. In this way, the determination of the LST is reduced to the determination of the critical path of each operation.

3.4 A Method of Generating the Reduced SAT Formula for Any JSSP

Figure 3 shows the algorithm that produces a reduced codification for the SAT representation of any JSSP. It is possible to check the satisfiability of the reduced codification obtained using SAT solvers.

In this algorithm, for each disjunctive arc (the resource capacity constraint), two clauses exist. One clause exists when operation *i* precedes *j,* and the other clause exists when the operation *j* precedes *i*. The generation of the RSF, is a function of the number of disjunctive arcs on the JSSP graph (see Fig 1). The number of disjunctive arcs are defined as *arc = m(n-1)*, where *m* is the number of machines and *n* is the number of jobs. For the purpose of comparing the complexity of the construction of RSF with the construction of the Crawford and Baker formula, the same number of jobs and machines are used so *m=n*. With this simplification, the RSF generates:

```
procedure reduced_codification (N,E,P)
{ N is a set of operations }
{ E is a set of disjunctive arcs without a sequence of use assigned}
{ P is a set of processing times of each operation }
{ The data represents one schedule as defined (feasible or not) }
{ C is the set of clauses in CNF, at the beginning C is empty }
begin
   for k=1 to number of arcs in E do
   begin
      { pr_{i,j} are arcs in E }
      C = C ∧ (~sa_{i,ri} ∨ ~pr_{i,j} ∨ sa_{j,di});
   end
   C is the reduced codification;
   return C;
end.
```

Fig. 3. The algorithm for constructing a reduced codification for the SAT representation of a JSSP

$$Clauses = 2n^2-2n \tag{1}$$

In addition, as each clause of the reduced SAT formula contains 3 literals, the evaluated literals are:

$$Literals=6n^2-6n \tag{2}$$

After taking into account the information in (1) and (2), the complexity in order to generate the RSF is $O(n^2)$.

In the case of the complete codification, when $m = n$, in addition to generating the RSF clauses, it is necessary to evaluate each $pr_{i,j}$ clause. For each pair of operations belonging to the same job, $n(n-1)$ clauses is needed. For each operation, it is also required to evaluate the following 5 types of clauses: $sa_{i,ri}$ (with one literal) that is equivalent to n^2 clauses for all operations, $eb_{i,di}$ (with one literal) that is equivalent to n^2 clauses for all operations, and three types of clauses $\sim sa_{i,t} \vee sa_{i,t-1}$, $\sim eb_{i,t} \vee eb_{i,t+1}$, $\sim sa_{i,t} \vee \sim eb_{i,t+pi-1}$ (each one with two literals) that are equivalent to $3n^2$ clauses for all operations. If all the clauses that need to be evaluated are added together, the Crawford and Baker formula generates:

$$Clauses = 8n^2-3n \tag{3}$$

The literals that they will evaluate are:

$$Literals=15n^2-7n \tag{4}$$

It can be seen from (3) and (4) that the complexity in order to generate the Crawford and Baker formula is $O(n^2)$. Although the complexity of generating the SAT formula in the two methods is the same, it is clear that evaluating RSF is simpler. The simplification is demonstrated in 2 and 4 where it can be seen that when using RSF it is necessary to evaluate the truth-value of a smaller number of literals.

4 Experimental Results

Several tests were performed in order to verify the reduction approach presented in this paper. In Table 6, the comparison of the number of clauses produced is shown. Two methods are examined, that of Crawford and Baker, and that of the reduced codification. The problems were taken from Beasley [19]: the problem, FT6, has 6 jobs and 6 machines, the problem, FT10, has 10 jobs and 10 machines, etc. The comparison of these results is shown in Table 6. The reduced codification (RSF) reduced an average of 75.9% of clauses for the nine problems.

Table 6. Number of clauses in a JSSP

Problem	Number of clauses		% reduction
	complete SAT codification	RSF	
FT6	270	60	77.78
FT10	770	180	76.62
LA21	3,465	840	75.76
LA24	4,536	1,106	75.66
LA25	4,925	1,200	75.63
LA27	5,751	1,404	75.59
LA29	6,641	1,624	75.55
LA38	11,438	2,812	75.42
LA40	12.680	3,120	75.39

5 Conclusions

RSF produces a significant reduction in the number of clauses (75.9%). The key to this reduction approach is the determination of the t times as the latest starting times (with a complexity of $O(N)$) of each operation for a proposed schedule. The complete SAT codification of Crawford and Baker, requires an additional calculation of the t times using a procedure not described by the authors. The complexity for this calculates could not be less than a complexity of a linear order. Although the complexity for the generation of the SAT formula in the two methods it is the same, it is clear that the evaluation of RSF is simpler because RSF requires the evaluation of the truth-values of a smaller number of literals. Due to the fact that RSF simplifies the number of clauses by such a significant percentage and the efficiency of linear complexity for obtaining the times t, it can be observed that it is more advantageous to use RSF than the complete SAT codification.

The algorithm that generates the RSF is applied only once in the beginning of the process of using a solver. RSF can be used in optimization methods that need initial solutions.

References

1. Garey, M. R., Johnson, D. S. and Sethi, R.: The Complexity of Flow Shop and Job Shop Scheduling, in Mathematics of Operation Research, Vol. 1, No. 2 (1976) 117-129

2. Conway, R.W., Maxwell, W.L. and Miller, L.W.: Theory of Scheduling. Addison-Wesley, Reading, Massachusetts (1967)
3. Smith, S.F. and Cheng, C.C.: Slack-Based Heuristics for Constraint Satisfaction Scheduling, in Proc. of the 11th National Conf. on Artificial Intelligence, Washington, D.C., (1993) 139-145
4. Ullman, J.D.: NP-complete scheduling problems, in Journal of Computer System Sciences, Vol. 10 (1975) 384-393
5. Crawford, J.M. and Baker, A.B.: Experimental Results on the Application of Satisfiability Algorithms to Scheduling Problems, in Proc. of the 12th National Conf. on Artificial Intelligence, Austin, TX, (1994) 1092-1098
6. Panwalker, S.S. and Iskander, W.: A survey of scheduling rules, in Operations Research, Vol. 25, No. 1, (1977) 45-61
7. Adams, E., Balas, E. and Zawack, D.: The Shifting Bottleneck Procedure for Job Shop Scheduling, in Management Science, Vol. 34, No. 3 (1988) 391-401
8. Schutten, M.J.: Practical job shop scheduling, in Annals of Operations Research, Vol. 83, (1988) 161-177
9. Carlier, J. and Pinson, E.: An algorithm for solving the job-shop problem, in Management Sciences, Vol. 35, No. 2 (1989) 164-176
10. Yamada, T. and Nakano, R.: Job-Shop Scheduling by Simulated Annealing Combined with Deterministic Local Search, in Metaheuristics Int. Conference, Colorado, USA, (1995) 344-349
11. Zalzala, P.J. and Flemming: Genetic algorithms in engineering systems, in A.M.S. Inst. of Electrical Engineers (1997)
12. Albert Jones and Luis C. Rabelo: "Survey of Job Shop Scheduling Techniques," NISTIR, National Institute of Standards and Technology, Gaithersburg, MD, 1998. In http://www.mel.nist.gov/msidlibrary/summary/authlist.htm.
13. Papadimitriou, C.H., Computational Complexity, Addison Wesley Pub. Co., USA, ISBN 0-201-53082-1, (1994).
14. Selman, B., Levesque, H. and Mitchell, D. A new method for solving hard satisfiability problems, In Proceeding of the Tenth National Conference on Artificial Intelligence, 139-144, 1992
15. Selman, B, and Kautz, II,: A. Local search strategies for satisfiability testing, In Procceding DIMACS Workshop on Maximum Clique, Graph Coloring and Satisfiability. 1993.
16. Davis, M., Logeman, G., and Loveland, D.: A machine Program for theorem proving. In CACM, (1962) 394-397.
17. Balas, E: Machine Sequencing via Disjunctive Graphs: An Implicit Enumeration Algorithm, in Operations Research, Vol. 17 (1969) 941-957
18. Garey, M.R. and Johnson, D.S.: Computers and Intractability: A Guide of the Theory of NP-Completeness, W.H. Freeman and Co, New York (1979)
19. Beasley, J.E.: OR Library, Imperial College, Management School, http://mscmga.ms.ic.ac.uk/info.html (1990)

Action Reasoning with Uncertain Resources

Alfredo Milani[1] and Valentina Poggioni[2]

[1] Dip. Matematica e Informatica, Università di Perugia
Via Vanvitelli 1, 06123 Perugia (Italy)
milani@dipmat.unipg.it
[2] Dip. di Informatica e Automazione, Università di Roma Tre
Via della Vasca Navale 79, 00146 Roma (Italy)
poggioni@dia.uniroma3.it

Abstract. In this paper we present RDPPlan, an automated problem solver which generates plans of actions in order to satisfy logical goals and numerical goals on uncertain resources. Planning with resources is a component of many applications which range from robot planning to automated manufacturing and automatic software composition. In the classical planning model the actions describe purely logical state transitions; some extensions have been recently proposed in order to manage numerical resources which can be produced/consumed in exact amounts. Unfortunately in real domains it is impossible to make accurate and exact previsions about resource production/consumption because of the inherent uncertainty of real world. The planning model introduced in RDPPlan allows to manage uncertainty about the initial value of resources and actions that can make uncertain updates of numerical resources. The proposed model uses the notion of trapezoidal fuzzy intervals to handle the uncertainty on resource values; the solving algorithm extends, for fuzzy resources, the propagation rules of the planner DPPlan.

Keywords: Artificial Intelligence, Planning, Fuzzy Sets, Problem Solving

1 Introduction

Automated Planning is a very important topic in artificial intelligence researches. The main objective is to solve problems in a domain world; the domain describes by a formal language some aspects, states and dynamics of the real world, where modifications are caused by actions. In particular given an initial world configuration (initial state) and a desired final configuration that we want to obtain (goal state), a planner is a system that chooses the necessary succession of actions (i.e. a plan) to apply in order to achieve the desired state from the initial state.
The first model proposed, the *Classical Planning Model* dates back to the 70es. This model assumes to work in a domain world that is finite, discrete, fully observable, deterministic and without exogenous events. Obviously these assump-

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 563–573, 2004.
© Springer-Verlag Berlin Heidelberg 2004

tions are very restrictive and it is of great importance to extend this model in order to use planning in effective real world applications.
For this reason in the last years several extensions to the classical model are been proposed. The proposed approaches for *Planning Model with Resources* try to model systems that can deal with more expressive constructs than the classical ones and provide the possibility to define numerical constraints and action effects on a finite set of numerical state variables. A lot of models have been proposed [3, 4,5,8,9,10,11,12,13], but most of these only can deal exact resources, that is they can manage resources (for instance: fuel, energy, money etc) and action effects (in this case effects are resource amounts that an action can increase or decrease when is applied) represented with numbers. This is a strong simplification of real situations because in general we do not know which is the exact value assumed by a resource in a specific world evolution. Often we have actions that consume or produce quantities varying in unexpected amounts or the resource initial values we assumed are affected by intrinsic errors. In real situations the quantities which are actually produced/consumed cannot be precisely determined (e.g. fuel consumption per km) with a few exceptions (e.g. completely filling a tank), but they can be realistically restricted to an interval (e.g. consuming between 3 and 3.5 liters of fuel per km).

In this work both a model representing world with uncertain resources and a system that implements this model, called RDPPlan, are proposed. The proposed model represents a simple form of uncertainty on resource values using *Trapezoidal Fuzzy Intervals*. This choice has two main motivations: first these fuzzy sets adequately represent the typical source of uncertainty in planning domains with resources; the second one is efficiency, i.e. a simple representation allows to have an actually computationally efficient system.

Section 2 provides notations and basic ideas underlying planning models; in Section 3 we introduce our ideas to model planning domain with uncertain resources, while in Section 4 we explain the algorithm and the technique used to realize the system; Section 5 focuses on the strategy used in the system to mix logical and numerical aspects of the decision system; Section 6 concludes with some ideas to extend model and system to other forms of uncertainty that can involve not only resources but also the logical aspect.

2 Automated Planning in AI

In *classical planning* we consider only worlds that are finite and discrete, fully observable, deterministic, and without exogenous events; in the last years these original assumptions are been relaxed in order to obtain systems that can solve problems closer to real problems.
In this direction *planning models with resources* are been introduced to extend the classical ones in order to realize planners able to solve more realistic problems; these models assume worlds having numerical resources that can be produced and/or consumed by the action applications.
Several planning models and systems for resources management that extend all

the most successful planner approaches have been proposed; among them it is worth noticing UCPOP–like [3], Graphplan–like [10,9,11,8], SAT–like [13,12], and also HTN based approaches [4]. In this section we present some basic notion about automated planning and the main definitions used both in classical and in extended planning models.

2.1 Classical Planning

Def. 1 *Planning Model.*
A classical planning model is defined by:
- a finite e discrete state space S*,*
- an initial state $s_0 \in S$*,*
- a set $S_G \subseteq S$ *of goal states,*
- action $A(s) \subseteq A$ *applicable in each state* $s \in S$*,*
- a transition function $f(a, s)$ *for* $s \in S$ *and* $a \in A(s)$*.*

STRIPS [6] is one of the oldest, simplest and most used planning languages. It is too restrictive so, in the last years, it has been changed and extended, but we still maintain its planning problem definition and action representation. Now, the standard language for domain and problem definition is PDDL 2.1 [7] that provides issues to manage numerical resources. The first version of PDDL was introduced and used for the first time in the first planning competition organized in 1998.

Def. 2 *Planning Problem.*
A classical planning problem *P is defined by a triple* $P = (I, G, O)$*, where I is the logical description of the facts which are true in the* initial state*, G is a set of facts that we want to be true in the* final *or* goal state*, O is a set of* operators *which can transform state into state; an action is a fully instantiated (ground) operator.*

Def. 3 *Operators.*
An operator $o \in O$ *is represented by three lists:*
- the Add(o) list *encodes the atoms that o make true,*
- the Del(o) list *encodes the atoms that o make false,*
- the Pre(o) list *encodes the preconditions of the operator, such as the atoms that must be true for o to be applicable.*

Def. 4 *Solution Plan.*
A solution for a planning problem P is a sequence of applicable actions that maps the initial state I into a goal state $s \supseteq G$*.*

2.2 Extending Classical Planning with Resources

Planning with Resources models, in addition to the logical relationships among domain objects, operators and states, can handle quantitative aspects of the

world, such as operators which involve consumable/reusable resources, domain constraints on resources and goals involving quantities. In general they extend the classical model presented in Def.1 using a new definition for states [9]. Let $r_1, r_2, \ldots, r_n$ be the resources present in the domain, then we have

Def. 5 *States with Resources.*
A state *is a couple* $s_t = (p(s_t), v(s_t))$*, where* $p(s_t)$ *is the set of propositions characterizing the state and* $v(s_t) = [v_1(s_t), \ldots, v_n(s_t)] \in \mathbb{R}^n$ *is a vector of real numbers where each* $v_i(s_t)$ *is the value of* r_i *in the state* s_t *(state at the time t).*

The operator definition is extended with two vectors that encode preconditions and effects on resources. The preconditions are expressed as constraints between numbers and the logical operator $\wedge$ that can join two conditions in order to create an interval constraint. In this way, for example, we use $(r \geq v_1) \wedge (r \leq v_2)$ to indicate the constraint $r \in [v_1, v_2]$. The effects can increase, decrease or assign values to resources.

Def. 6 *Operators with Preconditions and Effects on Resources.*
An operator $o \in O$ *is represented as in Def.3 plus two vectors of expressions:*
- Res_Eff(o)=$[\langle op \rangle\, r_1\, val_1, \ldots, \langle op \rangle\, r_n\, val_n]$*, where* $\langle op \rangle = \{+ | - | = | no\}$ *('no' states that operator not change the corresponding resource) and* $val_n \in \mathbb{Q}$*,*
- Prec_Eff(o)=$[\langle cond \rangle\, r_1\, val_1, \ldots, \langle cond \rangle\, r_n\, val_n]$*, where* $\langle cond \rangle = \{\leq | < | \geq | > | = | no\}$ *('no' states that operator has not condition over the corresponding resource) and* $val_n \in \mathbb{Q}$*.*

With these definitions for states and operators the planning problem definition in Def.2 become

Def. 7 *Planning Problem with Resources.*
A planning problem with resources *is defined by a triple* $P_r = (I_r, G_r, O_r)$*, where* I_r *is a state with resources that contains both the logical description and the resource values for the initial state,* G_r *contains the logical goals and a vector of conditions over resources that represents the set of explicit goals over resources and* O_r *is the set of operators with preconditions and effects over resources.*

The solution plan definition is the same of Def.4 where we intend as *executable actions* all that actions that have both logical and resource preconditions satisfied.

3 An Extension of the Planning Model with Resources

Models as [8,9,10,11] certainly represent an important step toward a more accurate model of the real world, but most of these fail to give any account of the potential imprecision which can affect the quantities related to resources. Many facts in the real world can be described in a satisfactory way by a boolean proposition, but it is not very realistic to assume that an exact number can model the continuous quantities describing a given resource.

In this section we present our ideas about the numerical framework that can extend the models presented in Section 2.1 and in Section 2.2. The basic idea is to extend the model in order to deal a simple kind of uncertainty about resource quantities.
A first step is the same direction was made in [12] where a rough representation of indecision is proposed. The authors represent resource values with real intervals, and use a simple interval algebra to construct the solution.

In this paper we propose an extension using *Fuzzy Intervals* to represent an actual uncertainty over resource values and operator effects. Fuzzy intervals allow to model adequately the imprecision of most resources managed in planning domains. The levels of typical planning resources (such as power supply, fuel, money etc.) are varying over intervals in a fairly continuous way, while they are produced or consumed by the execution of planning actions. In particular *Trapezoidal Fuzzy Intervals* are an effective way to represent resources bounded in an interval with fuzzy borders and, moreover, offer computational advantages with respect to an approach using generic fuzzy intervals or sets having a computationally hard membership function.

In our framework we represent a planning problem with a tuple $(I, G, O)_f$ that is another extension of the classical one proposed in Def.2, where each element is appropriately extended in order to handle fuzzy resources in preconditions, effects, initial state and goals. For notation simplicity we refer to a *Trapezoidal Fuzzy Interval* with the abbreviation TFI.

3.1 The Model

The resource model should represent the resource evolution over time and allow to reason on resource values in order to compute the solution plan. For these reasons we have defined a function that maps each pair (r, t) with the TFI that represents the possible values of resource r at the time t: $Val(r, t) = (a_1, a_2, a_3, a_4)$.

Def. 8 *State with Fuzzy Resources.*
A state with fuzzy resources is a couple $s_t = (p(s_t), V(s_t))$*, where* V *is the vector of* $Val(r_i, t)$*,* $\forall r_i$ *in the domain.*

Using the function Val we can define the *Initial State* of a resource r as $Val(r, 0)$.

Action preconditions and effects over resources are also represented by $TFIs$; the operator definition is the same of Def.6 where, in this case, the vector V has values defined by 4-ple of real numbers. For readability reason in the following we use $Pre(r, o)$ and $Eff(r, o)$ to represent, respectively, the precondition and the effect of the operator o over resource r. When an action A is applied in a state s_t, its effects define the new state s_{t+1}, called successor state of s_t.
The *add* and *delete* lists define the logical effects as in the classical model; new values for resources are instead obtained adding, subtracting or assigning (according to the cases) $Eff(r, A)$ to $Val(r, t)$ for each involved resource; the values representing each resource r in the successor state are defined by the function $Val(r, t + 1)$.

Def. 9 *Fuzzy Resource Constraints*
A constraint on fuzzy resources can be represented by expressions as in Def.6 and by expressions as $(\langle cond \rangle\, r_i\, f_set)$
where $\langle cond \rangle = \{< \mid \leq \mid = \mid > \mid \geq \mid no\}$ *('no' states that operator has not condition over the corresponding resource)* f_set *is a TFI and* r_i *is a domain resource.*

Conditions as in Def.6 are translated into (eventually infinite) fuzzy set F. For example a condition like $(\geq\ fuel\, 8)$ is represented by $F = (8, 8, \infty, \infty)$. The constraint have to be satisfied for all possible resource value in the set defined by function Val. It is possible to ensure this for a time t when is verified the condition

$$Val(r, t) \subseteq F \tag{1}$$

The second type of constraints also require that the conditions are satisfied for each possible value of the resource. Then, let $Val(r, t) = (v_1, v_2, v_3, v_4)$ and $f_set = (f_1, f_2, f_3, f_4)$ we say that $(\leq\ r\, f_set)$ is satisfied at time t if and only if $v_4 \leq f_1$ and that $(\geq\ r\, f_set)$ is satisfied at time t if and only if $v_1 \geq f_4$; the meaning of "=" condition is straightforward.

Constraints between fuzzy resources are used both in action preconditions and in resource goals. The definitions of action executability and goal satisfiability require that all resource constraints and all logical conditions expressed in action preconditions and goal definition are satisfied.

The TFI representation is very useful for an efficient implementation of this model. The precondition/goal condition in (1) is very simple to check: it requires to check four linear conditions between real numbers. If $Val(r, t) = (v_1, v_2, v_3, v_4)$ and $f_set = (f_1, f_2, f_3, f_4)$ (1) become

$$\begin{cases} v_1 \geq f_1 \\ v_2 \geq f_2 \\ v_3 \leq f_3 \\ v_4 \leq f_4 \end{cases} \tag{2}$$

3.2 Simultaneous Actions

The model proposed admit the possibility of *Simultaneous (Parallel) Actions*. From a definition of parallel actions as in GraphPlan [2], it is possible to compute shorter plans (plans requiring less time) because the executor can apply more than one action at each time instant. In GraphPlan two or more actions are defined *Simultaneously Executable* if they are all executable and they give the same result, independently from execution order.
If we want translate this definition for a model with resources we must check that each execution order is possible, that is the effects over resources not affect the possibility of execution of another action having preconditions over the same resource, and we must check that the actions are in some sense "commutative" with respect to resources.

As a straightforward consequence of the second condition, an assignment on resource r is not simultaneously executable with any other action changing r. The question remains open whether to allow an action changing r to be simultaneous with any action having a precondition with respect to r. Our approach is to allow simultaneity whenever the change does not affect the executability. In the following we called additive, the actions having increase or decrease effects.

In order to apply m actions at the same time t it is necessary to compute with respect to a resource r the fuzzy set that can allow to do it; this set is called *Simultaneity Fuzzy Set* and it is denoted by $Sim_Set(\{A_1, \ldots, A_m\}, r, t)$.

Def. 10 *Actions Simultaneously Executable. If* $Val(r,t) \subseteq Sim_Set(\{A_1, \ldots, A_m\}, r, t)\, \forall r$ *then* $A_1, \ldots, A_m$ *are simultaneously executable.*

Proposition 1 *Let be* $Pre(r, A_i) = (e_{1i}, e_{2i}, e_{3i}, e_{4i})$ $Eff(r, A_i) = (f_{1i}, f_{2i}, f_{3i}, f_{4i})$, *for* $i = 1, \ldots, m$.
We have that the action A_i*'s are simultaneously executable (with respect to* r*) if* $v_1 \leq v_2 \leq v_3 \leq v_4$, *where*
$v_1 = \max\{\alpha_{1i}, \alpha_{1i} - {k_i}^{--} : i = 1, \ldots, n\}$, $v_2 = \max\{\alpha_{2i}, \alpha_{2i} - {k_i}^{-} : i = 1, \ldots, n\}$,
$v_3 = \min\{\alpha_{3i}, \alpha_{3i} - {k_i}^{+} : i = 1, \ldots, n\}$, $v_4 = \min\{\alpha_{4i}, \alpha_{4i} - {k_i}^{++} : i = 1, \ldots, n\}$,
and $\forall i = 1, \ldots, n$
${k_i}^{++} = \sum_{j \neq i, e_{4j} > 0} e_{4j}$, ${k_i}^{+} = \sum_{j \neq i, e_{3j} > 0} e_{3j}$, ${k_i}^{--} = \sum_{j \neq i, e_{1j} < 0} e_{1j}$, ${k_i}^{-} = \sum_{j \neq i, e_{2j} < 0} e_{2j}$. *In the positive case* $Sim_Set(\{A_1, \ldots, A_m\}, r, t) = (v_1, v_2, v_3, v_4)$

The terms v_1, v_2, v_3, v_4 can be computed in linear time (in the number of resources and actions) by storing (and keeping updated) the values
${k_0}^{--} = \sum_{e_{1j} < 0} e_{1j}$, ${k_0}^{-} = \sum_{e_{2j} < 0} e_{2j}$, ${k_0}^{+} = \sum_{e_{3j} > 0} e_{3j}$, ${k_0}^{++} = \sum_{e_{4j} > 0} e_{4j}$.

The simpler case of simultaneous execution of additive actions on resource r (possibly none) with other actions which do not change r is solved defined as *Simultaneity Fuzzy Set* the intersection of all the precondition intervals.

A case not yet covered is the simultaneous execution of an assignment on r, say A_1, whose assigns to r the TFI (e_1, e_2, e_3, e_4), with actions $A_2, \ldots, A_n$ which do not change r. It is easy to see that $A_1, A_2, \ldots, A_n$ are simultaneous executable (with respect to r) if the intersection of all the precondition intervals is not empty and contains (e_1, e_2, e_3, e_4).

4 The RDPPlan Planning System

In this section we present the system RDPPlan that is the planner that implements the model previous described. It is derived from DPPlan [1], a purely logical GraphPlan-style planner based on propagation rules. First we will introduce the notion of *Realized Interval* and *Desired Interval*, the structures used to represent the resource states and goals; then we will show propagation rules and finally the strategy defined for achieving the goals over resources.

4.1 Structures and Propagation Rules

Given a planing domain having n resources and a planning graph structure having T levels (T is the last), we define a Txn matrix of pairs of TFI. Each pair is defined by (D_{rt}, R_{rt}) and the that elements are respectively called *DesiredInterval* and *RealizedInterval* for the resource r at the time-step t.
The interval R_{rt} represents the current value of the resource r at time t and it is computed considering the effects of the actions since now inserted in the plan; the other interval D_{rt} contains all the admissible values for the resource r that allow the execution of all the actions selected at time level t and it is computed taking into account the preconditions of the actions chosen at the time-step t.

Initialization phase
At the start of search procedure, for each resource r:
- for every time $t < T$, $R_{rt} = Val(r, 0)$,
- for every time $t < T$, $D_{rt} = (-\infty, -\infty, +\infty, +\infty)$
- $D_{rT} = G_r$.

Propagation phase
When a new action A having effects over a resource r is chosen for the solution plan at time level τ, each R_{rt} is updated according the following simple rules:
- if A has an assign effect `(= r` $Eff(r, A)$`)`, then $R_{r,\tau+1} = Eff(r, A)$
- if A has an increase effect `(+= r` $Eff(r, A)$`)`, then $R_{r,\tau+1} = R_{r\tau} + Eff(r, A)$
- if A has a decrease effect `(-= r` $Eff(r, A)$`)`, then $R_{r,\tau+1} = R_{r\tau} - Eff(r, A)$.

For time levels $t > \tau$, R_{rt} will be updated taking into account the effects of actions already chosen in these levels; in particular the updating is done a each time level until the one that has an assignment selected.
When a new action A is chosen for the solution plan, if A has precondition involving a resource r, a new D_{rt} is computed:
- if A is the first action chosen that involves r in preconditions or effects at the time-step t, then $D_{rt} = Prec(r, A)$
- if we have already chosen $A_1, \ldots, A_k$ that involve r at the time-step t, we have to compute the simultaneity set according to Prop.1 and set $D_{rt} = Sim_Set(\{A_1, \ldots, A_k, A\}, r, t)$.

4.2 Solution Plans and Action Executability

Def. 11 *Executable Plan.*
The sufficient and necessary condition for a plan to be executable and to be a solution of the given planning problem is that

$$R_{rt} \subseteq D_{rt}, \quad \forall r \quad \forall t. \tag{3}$$

5 Search Procedure Strategy

At each step, the search algorithm chooses a new action to try to add to the solution plan. DPPlan chooses the new action using a strategy that defines the best action for the moment. These strategies were defined to reduce the search space and the computational time.
RDPPlan works in a similar way. Since RDPPlan incorporates a framework for resources, it should be able to choose actions also according to criteria which take into account of resource values. This feature is necessary if we want solve problems having only resource goals. In general it is always useful: if we have no more logical goals to solve, but the problem is not solved because resource conditions are not satisfied, either we make a casual choice or we have a specific strategy.
We can notice that the width of a TFI can only grow when we consider additive operations and that it can decreases only when an assignment is applied. If we look at the R_{rt} width represents as, in some sense, the indetermination on resource r value, we have that the larger the interval width, the larger the indetermination is. Moreover note that if the width of the realized interval is large, it is more difficult that the solution condition (3) will hold.

For each time step t and for each resource r, we check if the condition $R_{rt} \subseteq D_{rt}$ holds: the negative cases are the goals on resources. Let $|R|$ denote the greatest width of TFI $R = (a_1, a_2, a_3, a_4)$, i.e. $|R| = a_4 - a_1$. The first control to do is on the widths of realized and desired intervals.
- If $|R_{rt}| > |D_{rt}|$ an assignment which assigns an interval with width less than $|D_{rt}|$ is the only possible choice. If there are many such assignments, the algorithm chooses that one which assigns the interval with the least width. If there are no assignments with this property, a backtracking is necessary.
- If $|R_{rt}| \leq |D_{rt}|$ the algorithm tries to solve this goal using this procedure:
- the algorithm searches for an action that can achieve the goal in one only step preferring, in the case of many available options, the action situated at the level nearest to t. In this way it is unlikely that an action which can negate the goal is selected.
- if this first search fails, the algorithm chooses an action that can permit to come closer to the goal. The choice criterium, now, take into account the distance between D_{rt} and R'_{rt} and their widths.

This criterium has been implemented by defining two preference functions, one for the interval widths and one for the distance between intervals, and by searching for an action that maximizes a linear combination of the functions. The first function is

$$f_W(A, D_{rt}) = \frac{|D_{rt}| - |R'_{rt}|}{|D_{rt}|} \tag{4}$$

and gives to each action A a numerical positive score between 0 and 1. The second function is a decreasing function of the distance between the middle points of the two intervals

$$f_D(A, D_{rt}) = \exp\left(-\frac{1}{2}\left|\sum_{\alpha=1}^{4}\alpha_i - \sum_{a=1}^{4}a_i\right|\right) \quad (5)$$

where $(\alpha_1, \alpha_2, \alpha_3, \alpha_4) = R'_{rt}$ and $(a_1, a_2, a_3, a_4) = D_{rt}$. Also this function gives to each action A a numerical positive score between 0 and 1. The first experimental results show that good values for the ratio c_1/c_2 of the coefficients of the linear combination
$f(A, D_{rt}) = c_1 f_W(A, D_{rt}) + c_2 f_D(A, D_{rt})$
are between 1 and 2.

6 Conclusions

In this work we have presented a model to generate plan of actions in domains having productable/comsumable resources. We can handle, in particular, cases in which the resource values are only partially known and/or actions can produce/consume uncertain amounts of resources.

The presented RDPPlan planner implements the introduced resource model. RDPPlan search algorithm make use of a strategy for action selection based on resource values, which represent a relevant original contribution of this work.

The semantics of the presented model makes extensive use of the notion of trapezoidal fuzzy intervals as a computationally feasible representation which adequately applies to most resources in planning domains.

Future research work will investigate domain dependent action selection strategies in order to increase the system efficiency. Other promising ongoing research aims at extending the model in order to manage other kinds of uncertain knowledge in planning domains, such as probabilistic knowledge, and to integrate the planning and execution phases in a resource framework.

References

1. Baioletti M., Marcugini S. and Milani A. (2000). DPPlan: An Algorithm for Fast Solution Extraction from a Planning Graph. In Proceedings of AIPS-2000, 2000.
2. Blum A., Furst M. (1997). Fast Planning Through Planning Graph Analysis. In *Artificial Intelligence*, 90(1-2), pag. 279-298, 1997.
3. Chien S. et al. (2000). ASPEN: Automated Planning and Scheduling for Space Mission Operation. In *Proceedings of SpaceOps 2000*, Colorado, USA, May 2000.
4. Currie K., Tate A. (1991). O-Plan: The Open planning Architecture. In *Artificial intelligence*, 52, pages 49-86, 1991.
5. Do M.B., Kambhampati S. (2001). SAPA: A Domain-Independent Heuristic Metric Temporal Planner. In *Proceedings of ECP-01*, pag. 109-120, 2001.
6. Fikes R.E., Nilsson N.J. (1971). STRIPS: A new approach to the application of theorem proving to problem solving. In *Artificial Intelligence*, 2(3/4), 1971.
7. Fox M., Long D. (2002) PDDL 2.1: An extension to PDDL for expressing temporal planning domains. 2002.
8. Haslum P., Geffner H. (2001). Heuristic Planning with Time and Resources. In *Proceedings of ECP-01*,pag. 121-132, 2001.

9. Hoffmann J. (2002). Extending FF to Numerical State Variables. In *Proceedings of ECAI-2002*, 2002.
10. Koehler J. (1998). Planning under Resource Constraints. In *Proceedings of ECAI-98*, pag 489-493, Brighton, UK, August 1998.
11. Refanidis I., Vlahavas I. (2000). Heuristic Planning with Resources. In *Proceedings of ECAI-00*,pag. 521-525, Berlin, Germany, August 2000.
12. RintanenJ., Jungholt H. (1999). Numeric State Variables in Constrained-based Planning. In *Proceedings of the Fifth European Conference on Planning, ECP'99*, Durham, UK, 1999.
13. Wolfam S., Weld D. (1999). The LPSAT Engine and its Application to Resource Planning. In *Proceedings of IJCAI-99*, pages 310-316, Stockholm, Sweden, August 1999. *Proceedings of AIPS-2000*, Breckenridge, CO, USA, April 2000.

Software Rejuvenation Approach to Security Engineering

Khin Mi Mi Aung and Jong Sou Park

Dept. of Computer Engineering,
Hankuk Aviation University, Seoul, Republic of Korea
{maung,jspark}@hau.ac.kr

Abstract. While traditional security mechanisms rely on preventive controls and those are limited in surviving malicious attacks, we propose a novel approach to security engineering. The objective is to characterize the attacks in real time and survive in face of attacks by using software rejuvenation. In this paper we address the critical intrusion tolerance problems ahead of intrusion detection. Firstly, the attacks are characterized by applying Principle Component Analysis (PCA) and these characterized intrusions are analyzed according to their state changes by utilizing transient state analysis. Subsequently, the software rejuvenation methods are performed by killing the intruders' processes in their tracks, halting abuse before it happens, shutting down unauthorized connection, and responding and restarting in real time. These slogans will really frustrate and deter the attacks, as the attacker can't make their progress. This is a way of survivability to increase the deterrence level against an attack in the target environment.

1 Introduction

An information system can face a variety of security terrorization during its lifetime and many of them may result in successful attacks, or intrusions. Security engineering is about building systems to remain dependable in the face of malice, error, or mischance [1]. It must include not only the ability to prevent attacks but also the ability to survive from and operate through attacks. If any attacks or intrusions can not solve them yet, those will be security-aging problems. The phenomenon of aging has been reported in heavy-duty systems and also in high-availability and safety-critical systems. To counteract this phenomenon, a proactive technique called "software rejuvenation" had been proposed [11]. This essentially involves gracefully terminating an application or a system and restarting it in a clean internal state. Thus, we apply software rejuvenation to be applicable for security engineering. As a consequence, the system survivability level will increase even under attacks. In the paper, we address the security engineering with Hybrid approaches in different aspects.

Firstly, the attacks are characterized within a short time interval by applying PCA [12] to the features of Internet packet flows. PCA is one of two classical approaches to find the effective linear transformations [6] and we use

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 574–583, 2004.
© Springer-Verlag Berlin Heidelberg 2004

it to project our data set onto a lower-dimensional subspace and to visualize the separate flows types. Short-term real time activities focus on specific tasks and recognize the accurate, abnormal behavior and up-to-date information in a timely fashion of short intervals, since we often need to update with new attack's response methods. PCA is a very fast linear transformation that achieves the maximum distance preservation when projecting from high to low dimensional feature spaces [16]. Cleveland et.al. [5] looked the statistical properties of Internet traffic and the difficulties of handling the complex and very large bases that result from collecting packet headers. They utilized the S statistical package for analyzing the data. Frank [8] applied artificial intelligent to perform these ideas and used the data collected by NSM. Heberlein et.al. [9] classified the flows from the Internet to local network including, flow duration, packets from source, packets from destination, bytes from source, bytes from destination and intrusion warning. Cannady [3] described a neural network trained to detect Network intrusions from packet header data. In our research, we second-hand a linear projection technique, called PCA, in order to reduce the density and predict the attacks. The statistical detection analysis provides most powerful features in intrusion detection [7,8,13]. It can be used for identifying trends in behavioral data and damage assessment.

As a next step, we analyze the characterized attacks according to their state changes in the transient period. It started from the system healthy state and finally it will return to system healthy state if we can successfully perform the rejuvenation, or else it will go through the system failure state. We perform the Software Rejuvenation methods to counter act these attacks' attempts or intrusions such as killing the intruders' processes in their tracks, halting abuse before it happens, shutting down unauthorized connection, and responding and restarting in real time. It is aimed to prevent the occurrence of more severe crash failures in the future and restoration of a robust system state. Our policy is to minimize the damage leakages and computing resources and to maximize the rewards. Rohlicek [15] proposed the critical role played by (almost) transient states which is resulting in a straightforward algorithm for the construction of a sequence of aggregate Markov generators associated with various time scales. Chen et.al, [4] proposed a quantitative approach to evaluate network survivability. They adopted system transient performance analysis to study the survivability performance of more complex systems. The transient state analysis in complicated systems described by a large system is usually time-consuming and thus costly. We have proposed the new approach of transient state analysis focus on applying software rejuvenation; using semi-Markov process and Weibull distribution function for survivability [2].

In the proposed architecture, we apply the hybrid approaches of multivariate data analysis, transient state analysis and rejuvenation methodologies and for the reasons of performance and quality, we will analyze the optimization as shown in (Fig. 1). In this paper we build the architecture for a process model of survivability. Our architecture is describing the prescriptions of the development of security engineering and combining with existing Firewall and Intrusion Detection Systems and let the system to be tolerated. We shall express how

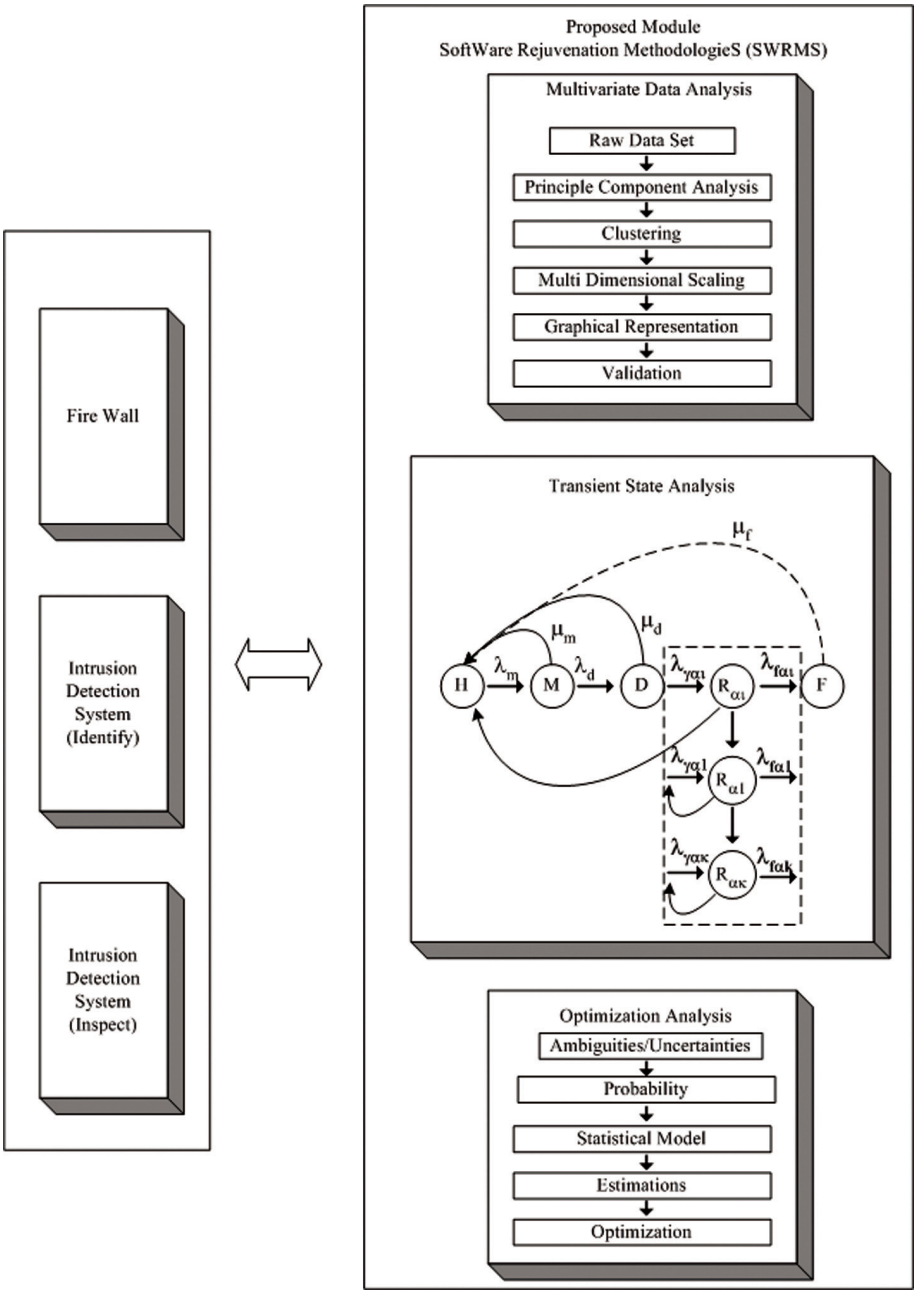


Fig. 1. Architecture of SWRMS

to depict the process by using appropriate sections. Next sections have been presented the utilized analysis trends with their respective experiments.

2 Principle Component Analysis (PCA)

We apply PCA for unsupervised learning of the normal runs and for extracting the symptoms of new attacks by minimizing their statistical profiles. Principal Component Analysis (PCA)'s linear transform has been widely used in data analysis and compression [14,16]. PCA is based on the statistical representation of a random variable. PCA offers a convenient way to control the trade-off between loosing information and simplifying the problem at hand. Solving eigen values and corresponding eigenvectors is a non-trivial task, and many methods exist.

By ordering the eigenvectors in the order of descending eigen values (largest first), one can create an ordered orthogonal basis with the first eigenvector having the direction of largest variance of the data. In this way, we can find directions in which the data set has the most significant amounts of energy. If we denote the matrix having the first eigenvectors as rows by, we can create a similar transformation $y = A_K(x - \mu), x = A_K^T\ y\ + \mu_x$.

By picking the eigenvectors having the largest eigen values we lose as little information as possible in the mean-square sense. One can for example choose a fixed number of eigenvectors and their respective eigen values and get a consistent representation, or abstraction of the data. We are here faced with contradictory goals: On one hand, we should simplify the problem by reducing the dimension of the representation. On the other hand we want to preserve as much as possible of the original information content. Now, consider a small example showing the characteristics of the eigenvectors. Eigenvectors and eigen values can be calculated from the covariance matrix.

3 Transient State Analysis

In this section, we analyze the characterized attacks according to their state changes in transient period. The state variables are Healthy State (H), Monitoring State (S), Detecting State (D), Rejuvenation State (R), and Failure State (F). Suppose a finite Markov chain with m states has some transient states. Assume the states are numbered so that T = 1, 2, ..., t is the set of transient states, and let P_T be the matrix of transition probabilities among these states. Let R be the t x (m-t) matrix of one-step transition probabilities from transient states to the recurrent states and P_R be the (m-t) x (m-t) matrix of transition probabilities among the recurrent states: the overall one-step transition probability matrix can be written as $P = \left\{ \begin{matrix} P_T & R \\ 0 & P_R \end{matrix} \right\}$. If the recurrent states are all absorbing then $P_R = I$.

The idea for Reset/restart technique is that can be used to recover a failed component while the process is resetting and/or restarting that component. Such expressions make the process would be performed on a standby component if the system has redundant components, or on the active component if it can be determined that it is a transient failure versus a hard failure. More specifically,

after the initial fault at the hardware layer, the controller/drive might retry x-times to eliminate transient and simple seek errors; then it might adapt the skew to attempt to recover the requested block. Ideally, we shall provide the ability to analyze trends in system performance as well as transient and recoverable events in order to predict faults. Our system is in passive monitoring mode that is genuine user traffic as it comes in and goes out. It also allows to sniff network packets and can be measured the response time and the packet loss percentages. It can also be viewed on Transmission Control Protocol (TCP) connection Establishment and Termination. We have studied transient state analysis focus on applying software rejuvenation; using semi-Markov process and Weibull distribution function for survivability in our previous work [2].

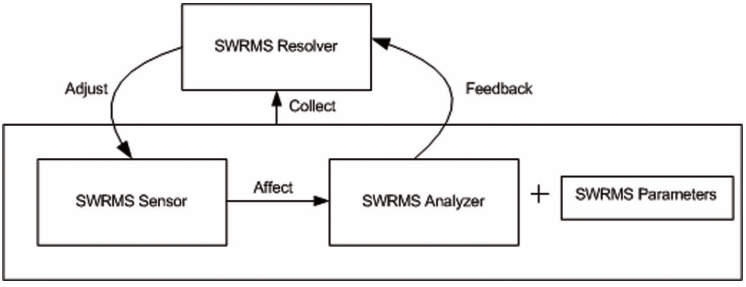


Fig. 2. Structure of SWRMS

4 SWRMS's Structure

The fundamental structure we address in the paper is conducted with four components (Fig. 2). To be characterized the attacks; SWRMS sensor shall perform an adequate monitoring process. SWRMS analyzer handles attacks detection and isolation. Based upon detection of the anomaly, the attacker can then be isolated from the network and prosecuted. As per experiences, SWRMS parameters have been formulated that could be maximized the system survivability. After collecting respective consequences, SWRMS Resovers launch the final decision to perform software rejuvenation process. As a first step, we processed the

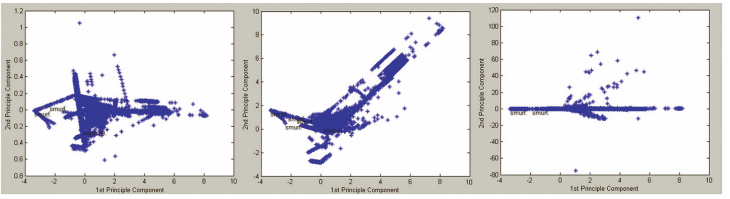


Fig. 3. PCA experiment results of extracted features in each specific attack

training sets and test sets suitable for the standard form of PCA. In second step, PCA results were under learning using the preprocessed learning sets; we validated a decision model by using the preprocessed validation sets. At last stage, we evaluated the detection rates of the decision model that had constructed in second phase by using test sets.

We computed unknown constraints by training the patterns from the first standard corpora for evolution of computer network intrusion detection systems, which has collected and distributed by MIT Lincoln Laboratory, under Defense Advanced Research Projects Agency (DARPA ITO) and Air Force Research Laboratory (AFRL/SNHS) sponsorship. These training patterns are the first formal, repeatable and statistically significant evaluations. Such evaluation efforts have been carried out in 1998 and 1999 and contributed significantly to the research field by providing direction for research efforts.

We use PCA for the dimension reduction because features are extracted with a linear transformation and our goal is the performance evaluation to fast density for unknown constraints. PCA has the preferred property of being simple, fast and repeatable. A new attack is one that has never been seen before. So we use simple matching (coordination level match) with our trained vector then we can estimate it. We extract the data set by using tcpstat [10] to analyze with PCA. tcpstat reports certain network interface statistics much like vmstat does for system statistics. It gets its information by either monitoring a specific interface or by reading previously-saved Tcpdump data from a file. We looked a sample data set that used fourteen different protocols and extract features with each specific attack (Fig. 3).

It is more for visualization that having any real basis and most similarity measures work about the same regardless of model. Rigorous formal model attempts to predict the probability that a given trace will be relevant to a given query. Ranks retrieved traces according to this probability of relevance (Probability Ranking Principle) and rely on accurate estimates of probabilities for accurate results. New attacks are being formulated everyday. The problem with detecting new attacks is that existing attack knowledge has to be aggregated with current behavior and events to determine that misuse is taking place. For the anomaly detection requiring multiple source pattern recognition associated with known misuse and the equivalent of instinct. (Fig. 4) illustrates the dataset with attack free and attacks. Our experimental results point out that the PCA transformations of every transient state could be evaluate the flow probability for each attack's features and network service density. For this purpose, our rejuvenation model tries to minimize the interdependency by closely estimate the performing time. Our resolvers provide the wide range of capabilities from security management configurations, log monitoring and network packet analysis. We associate with a specific resolver with each specific task. We set the set of rules associated with each specific resolver and each has predefined initial filter values, which are not concerned with other specific resolver. As an example, we can see that the connection must be in which state by using TCP resolver and the username successfully used during the authentication dialog as determined by the logging resolver. Our resolver's analyze based on the Tcpdump data.

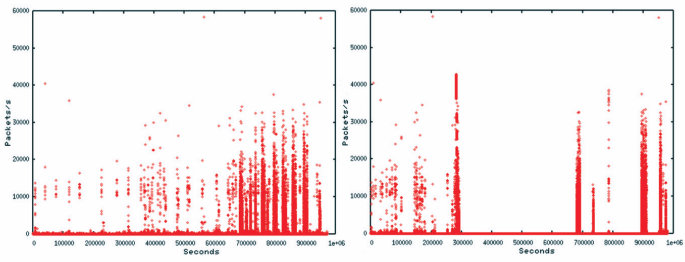


Fig. 4. Attack free and attacks data set

Tcpdump prints out the headers of packets on a network interface that match the Boolean expression. Since the intrusion detection system can't detect all malicious attacks, performing rejuvenation is more challenging for survivability.

We shall conduct an experiment of our resolvers process with TCP header and its sequence numbers. By transient state analysis, we can compute the rejuvenation rate, rejuvenation time, system fault rate, and recover rate. Our approach of operation models is a proactive and designed to find problems before they start. So that we can alert the users in advance before we are performing our approach, to prepare for their current processing tasks. The basic idea is to develop an operation model that maximizes security benefit and minimized total downtime cost.

Connections for TCP are well defined, because establishing and terminating a connection plays a central part of the TCP protocol unlikely with UDP. This transmission is termed as a request, even if in fact that application protocol being used is not based on request or relies. There are a number of generic procedures associated with TCP. We describe the experiment in next section.

5 Experiment

After analyzing the features of intrusions according to their state changes in transient period, we have found that the various Internet services have distinctive statistical signatures and it is possible to identify certain classes of service without content analysis. In this section, we present our resolver can detect retransmissions of a specific packet. Attackers always drop the retransmissions of a specific packet [7]. We show the two's complement because to form a-b we just add a to the two's complement of b. If TCP is expecting sequence number 1 and sequence number 6 arrives, since 6 is less than 1 using the sequence number arithmetic we showed, the data byte is considered a retransmission of a previously received data byte and is discarded. But if sequence number 5 is received, since it is greater than 1 it is considered a future data byte and is saved by TCP, awaiting the arrival of the missing bytes 2,3, and 4 (assuming byte 5 is within the received windows). When a retransmission packet is lost, TCP goes back to slow start phase and exponentially back offs its retransmission timeout value

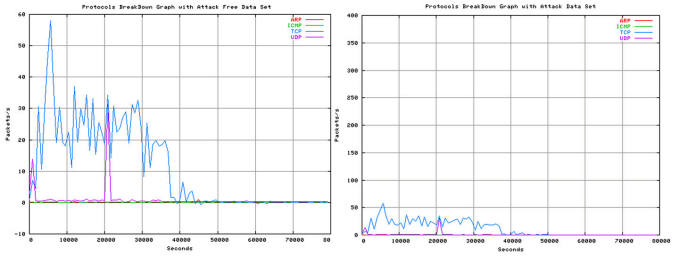


Fig. 5. Protocols breakdown graph with attack free and attacks data set

(RTO) upon every packet loss, with an upper limit of 64 seconds. We can infer that after a few consecutive retransmissions being dropped, the sender has to wait for a long period of idle time before performing a new retransmission.

Normally, no packets are sent out during this idle period. Thus, in [18], through NS2 simulation, retransmission packet dropping attacks can degrade the TCP's performance greatly by dropping only a few packets. In addition, since TCP connection gives up after sending retransmissions of a packet about 12 times, attackers can easily terminate the TCP service by dropping retransmissions. For the best-effort services, our packet dropping resolver is a congestion management mechanism implemented at each intermediate node that decides, proactively drop the packets to reduce congestion and free up precious buffer space in face of attack. With regard to TCP, these sequence number comparisons determine whether a given sequence number is in the future or a retransmission. Generally, packet dropping attacks can impact a network service on the following several aspects: Delay: e.g., dropping the retransmissions of packets in a FTP connection will drastically increase the total file transfer time.

While the primary goal is to avoid or combat congestion, the designs can significantly affect application throughput, network utilization, performance fairness, and synchronization problems with multiple TCP connections. We express protocols breakdown graph with attack free and attacks data set in (Fig. 5). Proactive discard packet does not discriminate between the packets belonging to multiple TCP connections. This may lead to a situation where one TCP connection finishes transmitting a packet and is able to increase its window size, which in turn may cause the cell of another TCP connection to be dropped due to buffer overflow. The second TCP connection, upon a timeout, is then forced to reduce its congestion window. Thus, the first TCP connection is able to obtain an unfair share of the bandwidth. Therefore, the conditions under which a cell is dropped are: (Buffer > Threshold) and (a weight W(x) > Z) $W(x) > Z \frac{B - Threshold}{C - Threshold}$.

Where B is the buffer size and C is the total amount of cells in the buffer. In this case, the weight is compared to a dynamic factor since it varies with the queue utilization. As congestion increases, the factor reduces in value so the weight is compared to a smaller value. Therefore, more and more packets are dropped and only the connections with fewer cells in the buffer will not suffer

losses. We can perform the proactive management of security aging problems and provide time to perform more detailed analyzing attacks' signatures and profile and predict symptoms by computing this with our packet dropping resolver.

6 Conclusion and Further Work

According to the PCA and Transient analysis, the paper has investigated that the attacks are characterized within the short intervals and every transient state could be evaluating the flow probability of each attack's features. And it is possible to categorize Internet traffic flows without content analysis. The paper has also forwarded the second related aspect of the operational needs for information systems such as the ability to operate through attacks, the ability to dynamically trade-off security, graceful degradation of non-critical functions in the face of intrusions and attacks when full functionality cannot be maintained, and performance and functionality as a function of the threat condition.

The awareness of the situations discovered in this paper is the first step in order to establish software rejuvenation approach to security engineering. Much work remains to be done in this direction. In fact, software rejuvenation does not remove bugs resulting from software aging but rather prevents them from manifesting themselves as unpredictable whole system failures. Periodic rejuvenation limits the state space in the execution domain and transforms a no stationary random process into a stationary process that can be predicted and avoided. Restricting intruder access through software rejuvenation actions is a short-term solution. Thus, this further research will be the longer-term solution by estimating the possibility of surviving under unknown attack by Analytic/Simulation method. Nonetheless, we believe that the basic ideas of software rejuvenation approach in this paper can be used to frustrate and deter the attacks. In this respect, the attacker can't make their progress and this is a way of survivability to increase the deterrence level against an attack in the target environment.

Our ongoing work is proceeding with the optimization analysis to increase the performance and quality. Our approach is only suitable yet with some attacks especially concerned with internal states, garbage collection, memory defragmentation, operating system kernel tables, and reinitializing internal data structures. In order to bring the system to provide cent per cent of critical functionality when under sustained attack, we will perform more work on probabilistically quantifying survivability with a Markov Chain's transition probabilities. New rejuvenation policies could be formulated based not just on time, but also on the symptoms. So our analysis raises opportunities and challenges by understanding the assumptions, the techniques, and their relationships, a designer of decision-making agents has many more tools with which to build the effective resolver's problems and the challenges lie in the development of additional tools and the integration of existing ones.

Acknowledgements. This research was supported by University IT Research Center(ITRC) Project and Internet Information Retrieval(IRC) Regional Research Center(RRC) Program.

References

1. Anderson, R.: Security Engineering:A guide to building dependable distributed systems. ISBN 0-471-38922-6, John Wiley and Sons USA (2001)
2. Aung, K.M.M.: The optimum time to perform software rejuvenation for survivability. In: Proc. of IASTED Int. Conf., SE 2004 : Innsbruck, Austria (2004) 292–296
3. Cannady, J.: Artificial neural networks for misuse detection. In: Proc. of NISSC Arlington, VA (1998) 443–456
4. Chen, D., Garg, S., Trivedi, K.S.: Network survivability performance evaluation: a quantitative approach with applications in wireless ad-hoc networks. ACM Press, Georgia, USA (2002) 61–68
5. Clevelnd, W.S., Sun, D.X.: Internet Traffic Data. Journal of the American Statistical Association (2000) 979–985
6. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. ISBN 0-471-05669-3 John Wiley and Sons USA (2000)
7. Ericsson, X.Z., Wu, S.F., Fu, Z., Wu, T.L.: Malicious Packet Dropping: How it Might Impact the TCP Performance and How We Can Detect It. In: Proc. of IEEE ICNP (2000) 263–272
8. Frank, J.: Artificial Intelligence and Intrusion Detection: Current and Future Directions. In: Proc. of the 13th NCSC Baltimore, MD (1994)
9. Heberlein, L.T., Dias, G.V., Levitt, K.N., Mukherjee, B., Wood, J., Wolber, D.: A Network Security Monitor. In: Proc. of IEEE SRCSP (1990) 296–304
10. Herman, P.: The tcpstat tool. http://www.frenchfries.net/paul/tcpstat (2003)
11. Huang, Y., Kintala, C.M.R., Kolettis, N., Fulton, N.D.: Software Rejuvenation: Analysis, Module and Applications. In: Proc. of FTCS-25 Pasadena, CA (1995) 381–390
12. Joliffe, I.T.: Principal Component Analysis. Springer-Verlag New York (1986)
13. Lee, W., Xiang, D.: Information-theoretic measures for anomaly detection. In: Proc. of IEEE Symposium on Security and Privacy, Oakland, CA (2001) 130–143
14. Oja, E.: Neural networks, principal components, and subspaces. International Journal of Neural Systems, 1(1) (1989) 61–68
15. Rohlicek, J.R., Willsky, A.S.: The reduction of perturbed Markov generators: an algorithm exposing the role of transient states. ACM Press 35(3) (1988) 675–696
16. Tian, Q., Moghaddam, B., Huang, T.S.: Display Optimization for Image Browsing LNCS issue 2184 ISBN: 3-540-42587-X, Amalfi, Italy (2001) 167–173
17. Wright, G.R., Stevens, R.W.: TCP/IP Illustrated. ISBN 0-201-63354-X Addison Wesley (1996)
18. Wu, T.L.: Securing Internet QoS: Threats and Countermeasures. Ph.D. Thesis, North Carolina State University (1999)

A Rollback Recovery Algorithm for Intrusion Tolerant Intrusion Detection System*

Myung-Kyu Yi and Chong-Sun Hwang

Dept. of Computer Science & Engineering Korea University,
1,5-Ga, Anam-Dong, SungBuk-Gu, Seoul 136-701, South Korea
{kainos, hwang}@disys.korea.ac.kr

Abstract. To cope with various intrusion patterns, an intrusion detection system, which is based on multiple agents working collectively, was proposed recently. Since an agent is easily subverted by a process that is faulty, a multi-agent based intrusion detection system must be fault tolerant by being able to recover from system crashes, either accidental or malicious activity. However, there have been very few attempts to provide fault tolerance in intrusion detection system. In this paper, we propose the rollback recovery algorithm for intrusion-tolerant intrusion detection system using communication-induced checkpointing and pessimistic message logging techniques. Thus, our proposed scheme guarantees a consistent global snapshot.

1 Introduction

The growth in Internet usage has been outpaced by increased security and fraud threats, which are increasing both in number and complexity. Enterprise network systems are inevitably exposed to the increasing threats posed by hackers as well as malicious users internal to a network. To resist these threats, intrusion detection has been an active research field for about two decades. Intrusion detection can be defined as the process and methodology of identifying and responding to malicious, inaccurate or anomalous activity targeted at computing and network resources[1,2]. Intrusion Detection System (IDS) successfully detect and trap attacks at a premature stage. To do so, it gathers data from network traffic, system logs, and application audit trails and reports on attacks to the system administrator when malicious or suspicious activity occurs.

Recently, numerous studies have focused on multi-agent based intrusion detection systems in order to detect intrusion behavior more efficiently[3,4,5,6,7, 8]. However, an agent is easily subverted by a process that is faulty. Multi-agent based IDS must be fault tolerant by being able to recover from accidental or malicious system crashes. If the IDS dose not have a general methodology for fault tolerance, IDS itself may ironically reinvite an attack from an attacker seeking to disable the IDS. One class of such attacks referred to as *clash attacks*

* This work was supported by grant No. R01-2002-000-00235-0 from the Basic Research Program of the Korea Science & Engineering Foundation

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 584–593, 2004.
© Springer-Verlag Berlin Heidelberg 2004

attempts to disable an IDS by causing it to fault or to run out of some critical resource. Assuming that it is infeasible to totally prevent these attacks, the goal of an IDS in the face of such an attack is to minimize the extend to which the attacker is successful in disabling the IDS. In our earlier work[6,7], we proposed a fault tolerant mechanism using uncoordinated checkpointing and pessimistic logging techniques. However, this approach has a drawback in terms of flexibility since each agent communicates and synchronizes only by exchanging messages through the agent coordinator. Also, it incurs a domino-effect[9]. In this paper, we propose a rollback recovery algorithm for multi-agents based IDS using pessimistic message logging and communication-induced checkpointing techniques. The rest of this paper is organized as follows. Section 2 illustrates the system model used in our proposal. Section 3 describes a new fault tolerant mechanism using communication-induced checkpointing. Section 4 addresses and discusses the correctness and presents simulation results. Finally, conclusions are presented in Section 5.

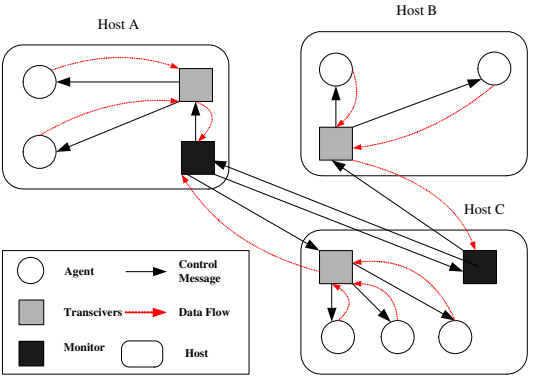


Fig. 1. The Architecture of our proposed system

2 System Model

2.1 System Description

Our proposed system architecture is similar to that in [4], but it has an additional function comparing with existing works. As shown in Fig. 1, each agent is an independently-running entity that performs *distributed computation* for certain aspects of a target host and reports doubtful activity to the appropriate transceiver. We define *distributed computation* as a collection of all communication events to detect intrusion behavior and agent failure of the target hosts. Generally, a transceiver or a monitor will generate an alarm for the user based on information received from one or more agents. By combining the reports from different agents, transceivers build a picture of host status, and monitors build a picture of network status. Agents communicate directly or indirectly with each

other in our proposed system architecture. A distributed computation is performed by a set of N agents, $\{A_1, A_2, \cdots, A_n\}$, running concurrently on target hosts in the network. Each agent has a sequence of state transition for its execution and the atomic action which causes the state transition is called an *event*. An event is *internal* if it causes no interaction with another agent. The message-sending and message-receiving events are *external* events. A sequence of events within an agent is called a *computation*. We assume that each ordered pair of agents is connected by an asynchronous, reliable, directed logical channel whose transmission delays are unpredictable but finite. Channels are not required to be FIFO (First-In First-Out). An agent can execute internal, send and delivery messages. An internal statement does not involve communication. When an agent A_i executes the statement *send(m)* to agent A_j, it puts the message m into the channel connecting A_i to A_j. When A_i executes the statement *receive(m)*, it is blocked until at least one message directed to A_i has arrived. Then a message is withdrawn from one of its input channels and delivered to A_i. Executions of internal, send and delivery statements are modeled by internal, sending, and delivery events.

Each agent runs on a different target hosts. Target hosts do not share a common memory, and there is no bound on their relative speeds. For simplicity, we assume that agents follow fail-stop behavior[9]. Each agent produces a sequence of events $e_{i,1}, ..., e_{i,s}, ...$ This sequence can be finite or infinite. Every agent A_i has an initial local state denoted $\sigma_{i,0}$. The local state $\sigma_{i,s}$(s>0) results from the execution of the sequence $e_{i,1}, ..., e_{i,s}, ...$ applied to the initial state $\sigma_{i,0}$. More precisely, the event $e_{i,s}$ makes A_i progress from local state $\sigma_{i,s-1}$ to local state $\sigma_{i,s}$. By definition, we say that "$e_{i,x}$ belongs to $\sigma_{j,s}$" (denoted $e_{i,x} \in \sigma_{i,0}$ if $i = j$ and $x \leq s$). Let H be the set of all events produced by a *distributed computation*. The distributed computation is modeled by the partially ordered set with $\widehat{H} = (H, \xrightarrow{hb})$, where $\xrightarrow{hb}$ denotes the well-known Lamport's **happen-before** relationship[9]:

$$e_{i,x} \xrightarrow{hb} e_{j,y} \Leftrightarrow \begin{cases} & i = j \wedge x \leq y \\ \vee & \exists m : e_{i,x} = send(m) \wedge e_{j,y} = receive(m) \\ \vee & \exists e : e_{i,x} \xrightarrow{hb} e \wedge e \xrightarrow{hb} e_{j,y} \end{cases}$$

2.2 Local Checkpoint and Global Checkpoint

A local checkpoint C is a recorded state of an agent. A local state is not necessarily recorded as a local checkpoint, so the set of local checkpoints is only a subnet of the set of local states.

Definition 1. *A communication and checkpoint pattern is a pair($\widehat{H}$,$C_{\widehat{H}}$) where $\widehat{H}$ is a* distributed computation *and $C_{\widehat{H}}$ is a set of local checkpoints defined by $\widehat{H}$.*

C_i^k represents the k-th local checkpoint of agent A_i and k is called the *sequence number* of this checkpoint. The local checkpoint C_i^k corresponds to some local state $\sigma_{i,s}$ with $k \leq s$.

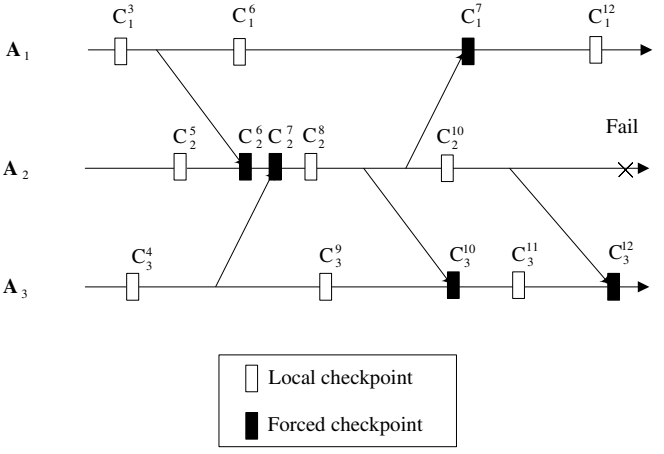


Fig. 2. Local Checkpoint and Forced Checkpoint

Fig. 2 shows an example of local checkpoint and forced checkpoint in our proposed system. We assume that each agent A_i takes an initial local checkpoint C_i^0(corresponding to $\sigma_{i,0}$), and after each event a checkpoint will eventually be taken. We say *send(m)* $\in C_i^k$ if message m was sent by agent A_i before taking the local checkpoint C_i^k. Similarly, we say *receive(m)* $\in C_i^k$ if message m was received and processed by agent A_i before taking the local checkpoint C_i^k. A global state is a collection of the individual states of all participating processes and of the states of the communication channel. Intuitively, a consistent global state is one that may occur during a fail-free, correct execution of a distributed computation, whereas inconsistent states occur because of failure. The state in Fig. 3 is an inconsistent state because agent A_k is shown having received m_2 but the state of A_j does not reflect sending it. Such a state is impossible in any failure-free, correct computation.

Definition 2. *A global checkpoint is consistent if all of its pairs of local checkpoints are consistent.*

3 The Proposed Rollback Recovery Algorithm

In this section, we propose a rollback recovery algorithm for multi-agent based IDS. Each agent performs *distributed computation* to detect intrusion behavior and agent failure of the target system. It also sends a monitoring message m

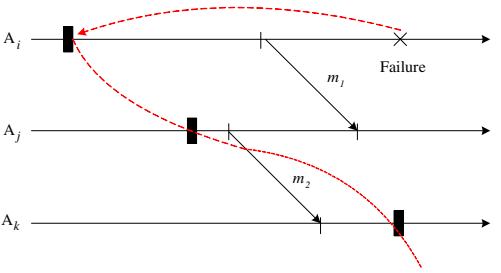


Fig. 3. An Example of a Inconsistent State

to the transceiver periodically. Sometimes multiple agents are required with each performing a single task. In that case, the transceiver receives monitoring messages from the several agents. To recover from agent failure, each agent takes local and forced checkpoints independently as follows.

Algorithm 1 : Agent_CheckPointing Procedure

Procedure Agent_Checkpointing{
  IF A_i sends a monitoring message m to the transceiver T_k **then** {
    $sn \leftarrow sn + 1$;
    $sflag \leftarrow 0$;
    Takes a local checkpoint C_i^{sn} with its sn ;
    It sends a monitoring message m to the transceiver T_k with C_i^{sn} ;
  }
  IF A_i sends monitoring message m to A_j **then** {
    $sflag \leftarrow 1$;
    It sends monitoring message m to A_j ;
  }
  IF A_i receives monitoring message m from A_j **then** {
    $sn \leftarrow sn + 1$;
    Takes a forced checkpoint C_i^{sn} ;
  }
  When A_i receives an ACK message from transceiver T_k {
    IF $sn < rsn$ **then**
    $sn \leftarrow rsn$;
  }
  It performs the tasks required for intrusion detection continually ;
}

Each transceiver has a variable gsn, and each agent has two variables sn and $sflag$. We denote sn as the sequence number of the latest local checkpoint taken by agent A_i. Also, rsn denotes the received sequence number from the transceiver. Finally, we denote gsn as the sequence number of the received latest local checkpoint from the agent at the transceiver and lsn denotes the

previous serial number of the last successful local checkpoint C_i^{sn} in failed agent A_i. The initial value of sn and gsn are equal to 0. Algorithm 1 shows the agent checkpointing procedure in each agent. When it is time for agent A_i to take a local checkpoint, it increases the value of sn and sets the value of $sflag$ to zero. Then, it takes a local checkpoint with sequence number sn and sends C_i^{sn} with its monitoring message m to transceiver T_k. Finally, A_i receives an acknowledge (ACK) message with rsn from transceiver T_k. If the value of rsn is larger than sn, A_i changes the value of sn to rsn. Whenever A_i sends monitoring message m to other agent A_j, it sets value of $sflag$ to 1. After receiving monitoring message m with C_i^{sn} from A_i, transceiver T_k performs the transceiver checkpointing procedure as shown in Algorithm 2.

Algorithm 2 : Transceiver_CheckPointing Procedure

```
Procedure Transceiver_Checkpointing{
   When transceiver T_k receives monitoring message m with C_i^sn from A_i then {
     rsn ← sn ;
     It compares the received sn from C_i^sn with its gsn ;
     If sn > gsn then {
       gsn ← sn ;
     }ELSE If sn ≤ gsn then {
       rsn ← gsn ;
     }
     Takes local checkpoint C_k^gsn ;
     It sends an ACK message to A_i with rsn
     Finally, it sends monitoring message m to its monitor ;
   }
}
```

Then, T_k sets the value of rsn to sn and compares the received sn from C_i^{sn} with its gsn. If the value of sn is larger than gsn, it sets the value of gsn to sn. Otherwise, it sets the value of rsn to gsn. After setting the value of gsn, it takes local checkpoint C_k^{gsn} and sends the monitoring message m to its monitor. Finally, transceiver T_k sends an ACK message to A_i with rsn. The value of gsn is used to keep the sequence number of the latest local checkpoints of the agents at the transceiver and helps in the progression of the consistent global snapshot. In Fig. 2, the initial sequence number sn of A_1, A_2, and A_3 is 1 and the current value of gsn of T_4 is 3. When A_1 takes a local checkpoint, its sn changes to 2, and sends monitoring message m with C_1^2 to T_4. Because gsn is larger than sn (i.e., $3 > 2$), T_4 sets the value of rsn to 3 and sends an ACK message with rsn. Finally, A_1 changes sn to 3 because rsn is larger than its sn (i.e., $3 > 2$).

Algorithm 3 : Asynchronous_Recovery Procedure

```
Procedure Failed_Agent_Recovery {
  When an agent A_i fails {
    A_i send message to transceiver with i and sflag ;
  }
  When failed agent is restarted {
    Transceiver sends RollbackRecovery(C_i^lsn) message to A_i ;
    A_i recovers its previous status ;
    A_i resumes operation after latest checkpoint using C_i^lsn ;
  }
}
Procedure Transceiver_Recovery(i, sflag) {
  When a transceiver receives message from failed agent A_i {
    It finds lsn from earliest local checkpoint C_i^sn ;
    IF sflag ≠ 0 then {
      gsn ← lsn ;
      It rolls back to checkpoint C_k^gsn ;
      All checkpoints beyond gsn are deleted ;
      It sends RollbackRecovery(gsn) message to all other agents ;
    }
  }
}
Procedure Unfailed_Agent_Recovery {
  When A_j receives RollbackRecovery(gsn) message from T_k {
    IF sn from its C_j^sn > gsn then {
      sn ← gsn ;
      It rolls back to checkpoint C_j^sn ;
      All checkpoints beyond C_j^sn are deleted ;
    }
  }
}
```

Algorithm 3 shows an asynchronous recovery algorithm for a consistent global snapshot in multi-agent based IDS. When an agent A_i fails, it sends a message to the transceiver with agent number i and $sflag$, which then finds the previous serial number lsn from the last successful local checkpoint C_i^{sn} for the failed agent. Then, the transceiver examines the value of $sflag$ from the received message for A_i. If $sflag$ is equal to zero, it implies that A_i does not send any messages to other agents after the last checkpoint C_i^{lsn}. Thus, transceiver T_k only sends a $RollbackRecovery(C_i^{lsn})$ message to failed agent A_i after failed agent A_i is restarted enabling A_i to recover its previous status and resumes its operation after the latest checkpoint using C_i^{lsn}.

Conversely, if $sflag$ is not equal to zero, it sets a value of gsn to lsn and rolls back to checkpoint C_k^{gsn} and all the checkpoints beyond C_k^{gsn} are deleted for all agents. Finally, it sends $RollbackRecovery(gsn)$ messages to all other agents as

well as sending a *RollbackRecovery*(C_i^{lsn}) message to failed agent A_i after failed agent A_i is restarted. Thus, all agents roll back to their latest local checkpoint where the sequence number sn is less than or equal to consistent global snapshot gsn. In Fig. 4, T_4 receives the value of 2 (i) and 1 ($sflag$) from failed agent A_2, and finds a value of 10 (lsn) from its checkpoints. Then, T_4 rolls back to checkpoint C_4^{10} and all the checkpoints beyond C_4^{10} (i.e., C_4^{11} and C_4^{12}) are deleted. Finally, T_4 changes the value of gsn to 10 and sends a *RollbackRecovery*(10) message to all other agents because the value of $sflag$ is equal to 1. As a result, A_1 and A_3 roll back to checkpoint C_1^6 and C_3^9. When failed agent A_2 was restarted, T_4 sends a message with C_2^{10} to A_2. Fig 4 illustrates the recovery line before and after the rollback.

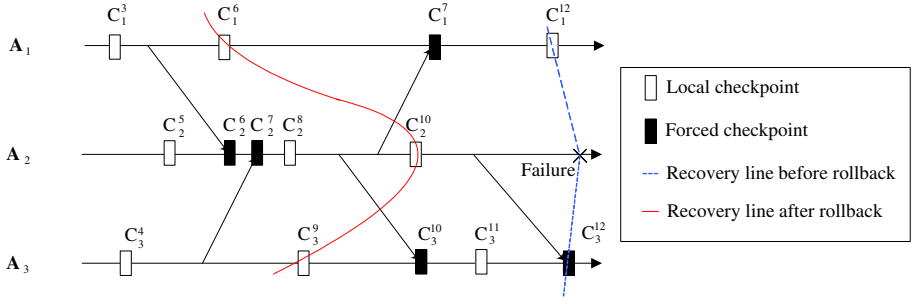


Fig. 4. Local Checkpoint and Forced Checkpoint

4 Discussion

4.1 Correctness

In this section, we prove the correctness of the proposed scheme. We observe that all the agents roll back to a consistent global checkpoint when an agent fails.

Observation 1 *When A_i receives a RollbackRecovery(gsn) message from T_k, it rolls back to checkpoint C_i^{sn} (where $sn \leq gsn$). Thus, the value of gsn is larger than or equal to sn. Also, the value of gsn is larger than sequence numbers of all checkpoints taken by A_i prior to checkpoint C_i^{sn}.*

Observation 2 *For any message m sent by A_i, if $sent(m) \in C_i^{sn}$ then sn is less than gsn. The converse is true. For any message m received by A_i, if $receive(m) \in C_i^{sn}$ then sn is less than gsn. However, the converse is true.*

Theorem 1. *When A_i sends a RollbackRecovery(gsn) message to all the other agents, all $A_j (i \neq j)$ roll back to a consistent local checkpoint $C_j^{sn} (sn \leq gsn)$ based above observations 1 and 2.*

4.2 Experiments and Analysis

In this section, we introduce the prototype we have developed as follows.

- Hardware
  Sun Blade 1000(2), Pentium 3(3)

- Operating System
  Sun Solaris 8, Redhat Linux 8.0, Window XP, Window 2000

- Programming Languages
  GNU C/C++ 3.2, J2SE 1.4.1,, Perl 5.004.02, tcpdump 3.6.2, nmap 2.5.4

- Other Development Tools
  mSQL 3.4, libpcap 0.6.2

The prototype we have developed is programmed through a combination of Java and C. All agents are implemented in C, and some agent use the *libpcap* library for packet capturing packets over network. For agent database and profile database, we employe *mSQL* database server to save agent status information and user profile. The module of transceiver and monitor are implemented in C and Java. User interface is implemented in *Perl*.

Table 1. Performance of the Proposed Communicated-Induced Protocol

Agent	λ	# of Local Ckps	# of Forced Ckps	Avg. Ckp. Size(MB)	Exec. Time (sec)
Agent 1	180	6	12	12.5	721
Agent 2	180	5	14	13.2	833
Agent 3	180	6	11	11.7	711
Agent 1	240	5	9	14.1	912
Agent 2	240	4	8	15.8	1021
Agent 3	240	4	10	12.3	755
Agent 1	360	2	4	14.4	951
Agent 2	360	2	7	16.2	1104
Agent 3	360	3	5	13.7	822

4.3 Simulation Results

The performance metrics we report are the number of forced checkpoints that a protocol causes and the performance overhead. Table 1 shows the results of the experiments, which consist of running each of three agents under the proposed communicated-induced protocol. Each agent triggers local checkpoints according to an exponential distribution with a mean checkpoint interval λ set to $180 \sim 360$ seconds. The results show that the average per-agent checkpoints size decreases as the frequency of checkpointing increases. Thus, the proposed communication-induced protocol has some of the negative performance properties of independent

checkpointing when used in computations where the agents are tightly coupled and communicate frequently.

5 Conclusions

In this paper, we proposed a new rollback recovery algorithm for multi-agent based IDS using communication-based protocol, which avoids the domino effect by piggybacking the control information to the application message. While agents are allowed to take some of their checkpoints asynchronously, agents may be forced to take additional checkpoints in order to ensure the progression of a consistent global snapshot. The decision to take the local checkpoint gsn is based on the received local checkpoint of each agent and the sending of an acknowledge message with rsn. In our proposal, each agent takes a local checkpoint as well as forced checkpoint when it receives a message from another agent. Using the sequence number and global sequence number, all agents roll back to a consistent global checkpoint when an agent fails. Thus, our proposed scheme guarantees a consistent global snapshot.

References

1. Dorothy. E. Denning, "An intrusion-dection model", *In Proc Symposium on Security and Privacy*, pp 118-131, 1986.
2. http://www.cerias.purdue.edu/coast/intrusion-detection
3. M. Crosbie and E.H. Spafford, "Active defense of computer system using autonomous agents", Technical report, COAST Group, Purdue University, February 15, 1995.
4. J. S. Balasubramaniyan, J.O. Farcia-Fernandez, D. Isacoff, E. Spafford, and D. Zamboni, "An architecture for intrusion detection using autonomous agents", Technical report, COAST Laboratory, Purdue University, June 11, 1998.
5. Ran Zhang, Depei Qian, Chongming Ba, Weiguo Wu, Xiaobing Guo,"Multi-agent based intrusion detection architecture", Proc of 2001 International Conference on Computer Networks and Mobile Computing, Page(s): 494 -501, 16-19 Oct. 2001
6. Myung-Kyu Yi and Chong-Sun Hwang,"Design of fault tolerant architecture for intrusion detection system using autonomous agents", Proc of the International Conference on Information Networking (ICOIN 2003), Jan 1, 2003.
7. Myung-Kyu Yi, Maeng-Soon Baik, and Chong-Sun Hwang,"Design of fault tolerant mechanism for multi-agent based intrusion detection system", Proc of the International Conference on Security and Management (SAM 2003), June, 2003.
8. Ran Zhang, Depei Qian, Chongming Ba, Weiguo Wu, Xiaobing Guo, "Multi-agent based intrusion detection architecture", *Proceedings. 2001 International Conference on Computer Networks and Mobile Computing*, pp494 -501, 2001
9. E. N. Elnozahy, D. B. Johnson and Y. M. Wang, "A survey of rollbalck-tecovery protocols in message passing systems," CMU Technical Report CMU-CS-99-148, June. 1999.

Design and Implementation of High-Performance Intrusion Detection System

Byoung-Koo Kim, Ik-Kyun Kim, Ki-Young Kim, and Jong-Soo Jang

Security Gateway System Team,
Electronics and Telecommunications Research Institute,
161 Gajeong-Dong, Yuseoung-Gu, Daejeon, 305-350, KOREA
{kbg63228, ikkim21, kykim, jsjang}@etri.re.kr

Abstract. The fast extension of inexpensive computer networks has increased the problem of unauthorized access and tampering with data. As a response to increased threats, many Network-based Intrusion Detection Systems (NIDSs) have been developed, but current NIDSs are barely capable of real-time traffic analysis on Fast Ethernet links. As network technology presses forward, Gigabit Ethernet has become the actual standard for large network installations. Therefore, there is an emerging need for security analysis techniques that can keep up with the increased network throughput. We have made effort to design and implement high-speed IDS that is run as a lower branch of our system named 'Network Security Control System (NSCS)'. Our IDS named 'Security Gateway System (SGS)' has a pattern matching approach through the FPGA (Field Programmable Gate Array) logic and kernel logic as detection mechanism that can be applied to Gigabit-Ethernet links. In this paper, we briefly introduce the whole architecture of our system designed to perform intrusion detection on high-speed links. And then, we present the efficient detection mechanism that is run by cooperation of FPGA logic and kernel logic. In other words, we focus on the network intrusion detection mechanism applied in a lower branch of our system.

1 Introduction

The fast extension of inexpensive computer networks has increased the problem of unauthorized access and tampering with data. As a response to increased threats, many Network-based Intrusion Detection Systems (NIDSs) have been developed, but current NIDSs are barely capable of real-time traffic analysis on Fast Ethernet links. As network technology presses forward, Gigabit Ethernet has become the actual standard for large network installations[1]. Therefore, there is an emerging need for security analysis techniques that can keep up with the increased network throughput. We have made effort to design and implement IDS that is run as a lower branch of our system named 'Network Security Control System (NSCS)'. Our IDS named 'Security Gateway System (SGS)' has a pattern matching approach through the FPGA (Field Programmable Gate Array) logic and kernel logic as detection mechanism that can be

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 594–602, 2004.
© Springer-Verlag Berlin Heidelberg 2004

applied to Gigabit Ethernet links. However, the real-time traffic analysis and high-speed intrusion detection is not an easy task. In this paper, we briefly introduce the whole architecture of our system designed to perform intrusion detection on high-speed links. And then, we present the efficient detection mechanism that is run by cooperation of FPGA logic and kernel logic. In other words, we focus on the network intrusion detection mechanism applied in a lower branch of our system.

The remainder of the paper is structured as follows. The next section presents the related works. Then, section 3 presents the architecture of our IDS system, and describes intrusion detection mechanism that we have applied in the proposed system. Section 4 introduces our prototype that we have developed. Finally, we conclude and suggest directions for further research in section 5.

2 Related Work

Basically, IDS is classified into Host-based IDS (HIDS) and NIDS. Audit sources discriminate the type of IDSs based on the input information they analyze. HIDS analyzes host audit source, and detects intrusion on a single host. With the widespread use of the Internet, IDSs have become focused on network attacks. NIDS uses the network as the source of security-relevant information. Consequently, NIDS moves security concerns from the hosts and their operating systems to the network and its protocols. Furthermore, the target of these NIDSs has been widened to address detection in large, complex network topologies. Therefore, more effective management strategies must be investigated for interoperability of IDSs scattered over widespread network. Although these management strategies are an important and interesting issue, it will not be discussed in this paper. Most of all, we focus on more effective and speedy detection strategies applied NIDS that performs intrusion detection independently[2].

In the last decade, networks have grown in both size and importance. In particular, TCP/IP networks have become the main means to exchange data and carry out transactions. But, the fast extension of inexpensive computer networks also has increased the problem of unauthorized access and tampering with data[2]. As a response to increased threats, many NIDSs have been developed to serve as a last line of defense in the overall protection scheme of a computer system. These NIDSs have two major approaches; misuse intrusion detection and anomaly intrusion detection[3][4], but most of existing NIDSs, such as Snort[5], NFR[6], and NetSTAT[7], only employs the misuse detection approach for reducing a lowering of performance to the minimum. Also, most of NIDSs based on misuse detection approach has concentrated on catching and analyzing only the audit source collected on Fast Ethernet links.

However, with the advancement of network technology, Gigabit Ethernet has become the actual standard for large network installations. Therefore, there is an emerging need for security analysis techniques that can keep up with the increased network throughput[1]. Existing NIDSs have problems of a lowering of performance as ever, such as bottleneck, overhead in collecting and analyzing data in a specific component. Therefore, the effort of performing NIDS on high-speed links has been

the focus of much debate in the intrusion detection community, and several NIDSs, such as RealSecure[8], ManHunt[9], and CISCO IDS[10], that is run on high-speed links actually has been developed. But, these NIDSs is still not practical because of technical difficulties in keeping pace with the increasing network speed, and real-world performance also will likely be less.

3 High-Performance Intrusion Detection System

In this section, we introduce the architecture of our system, named NSCS and components of the architecture. The architecture consists of two main components: SGS and Cyber Patrol Control System (CPCS). First, SGS performs the real-time traffic analysis and high-speed intrusion detection on two Gigabit Ethernet links that is run as inline-mode. Second, CPCS manages each SGSs and has a Policy Decision Point (PDP) functionality to each SGSs. Besides, CPCS has many functions for the whole system management. But, although the interoperability of overall system is an important and interesting issue, it will not be discussed in detail in this paper. In other words, we focus on the detection mechanism applied in a lower branch of our system.

3.1 System Architecture and Components

SGS is a substructure of CPCS aimed at real-time network-based intrusion detection based on misuse detection approach. As shown in the figure 1, SGS consists of three parts: Application Task block for communication channel with CPCS and system management functions, IDAB (Intrusion Detection and Analyzing Block) block for packet preprocessing and payload pattern matching, PSAB (Packet Sensing and Analyzing Block) block for packet sensing, preprocessor filtering, fixed field pattern matching and so forth. Again, we can divide IDAB block and PSAB block into several sub-modules. Most of all, the summary of the internal modules for detection operation is as following.

- PreProcessor Filter (PPF) module : checks out the incoming packet according to filtering rule, and decides which actual preprocessing function is necessary to be performed or not.
- Fixed Field Pattern Matching (FFPM) module : matches the incoming packet with fixed field patterns based on packet header information that is easily examined by fixed size and offset. Briefly, it performs the first pattern matching for detecting intrusions.
- PreProcessor (PP) module : performs the preprocessing function, such as protocol normalization, ip defragmentation, tcp reassembly and packet payload decoding, before step for payload pattern matching is run.
- Payload Pattern Matching (PPM) module : matches the first matching packet with payload patterns based on packet payload information that is not easily examined

by variable size and offset. Briefly, it performs the final pattern matching for detecting intrusions.

– Rule Manager (PM) module : manages the ruleset that is required for intrusion detection.

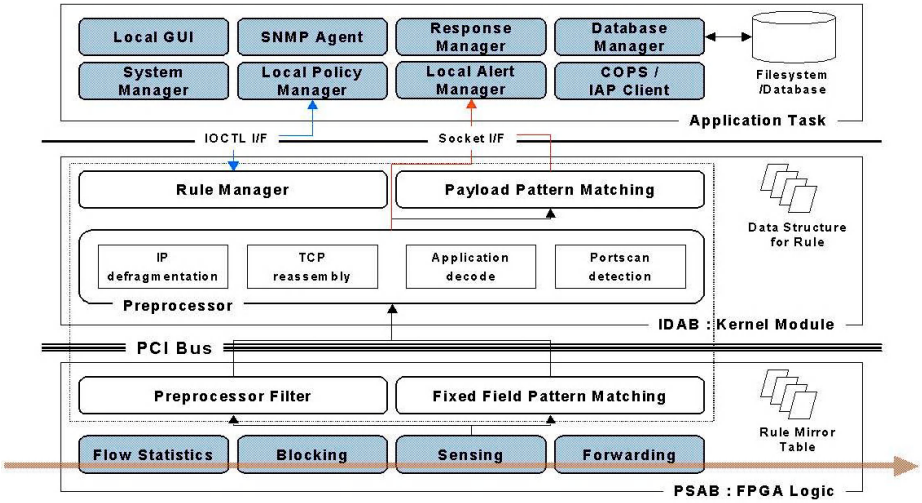


Fig. 1. System Architecture and Components

Through the interoperability of these components, SGS analyzes data packets as they travel across the network for signs of external or internal attack. That is, the major functionality of SGS is to perform the real-time traffic analysis and intrusion detection on high-speed links. Therefore, we focus on effective detection strategies applied FPGA logic and kernel logic.

3.2 Detection Rule Configuration

For detecting network intrusions more efficiently on high-speed links, our system divides its ruleset into two tables. As shown in the figure 2., one is Rule Mirror Table (RMT) for FPGA logic, and the other is Rule Table (RT) for kernel logic. First, RMT is configured to fixed field patterns based on packet header information that is easily examined by fixed size and offset. Therefore, it holds many common properties that must be included in each pattern. Second, RT is configured to payload patterns based on packet payload information that is not easily examined by variable size and offset. Therefore, it holds several properties that must be required for performing the payload pattern matching. Besides, RT holds several properties that must be included in generating alert message.

Detection rules of our system is configured to association of the above two tables, and their relationship is as following. First, the detection rules that applied to our system is divided into four groups according to a protocol value[11]: TCP group,

UDP group, ICMP group, IP group. In other words, each group has an association of RMT and RT that is configured to property values of the same protocol patterns. Therefore, each group has detection rules that are divided into fixed field patterns and payload patterns by RMT and RT. Basically, one fixed field pattern can have many payload patterns that are derived from its own. When incoming packets are being examined against a given detection rules, the packet is first compared along fixed field patterns in the RMT until the packet matches a particular fixed field pattern. Only if such a match occurs is the packet then compared along the payload patterns derived from the matching fixed field pattern. That is, detection rules of our system is managed and configured in the direction for reducing a lowering of performance by the packet processing in kernel logic to the minimum.

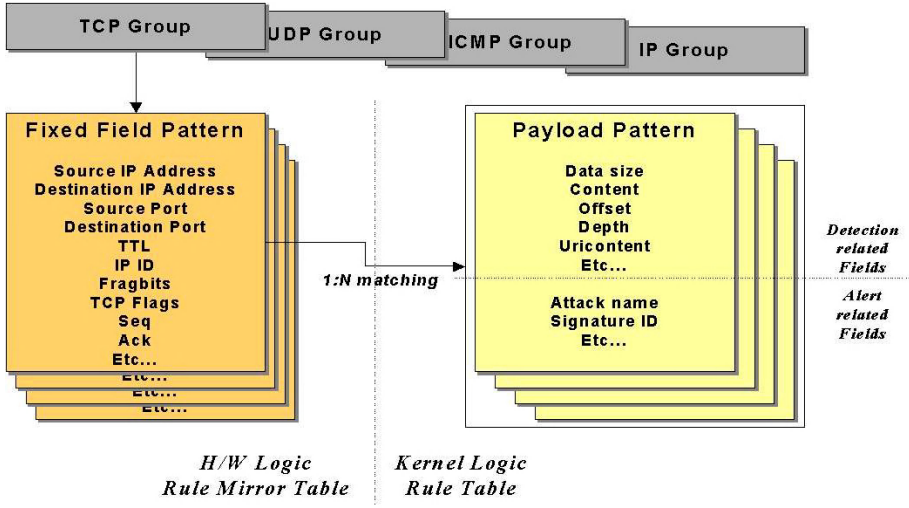


Fig. 2. Detection Rule Configuration

3.3 Detection Mechanism for High-Speed Intrusion Detection

Our IDS has a pattern matching approach through the FPGA logic and kernel logic as detection mechanism. First, the major functionality of FPGA logic is to perform the fixed field pattern matching and preprocessor filtering about incoming packets. And, It mainly is performed by PSAB block in components of SGS. Therefore, detection algorithm of PSAB block is very important as first step for intrusion detection.

As shown in the figure 3, FPGA logic has two packet processing flows performed concurrently. One is the flow for fixed field pattern matching, and the other is the flow for preprocessor filtering. First, the flow for fixed field pattern matching is as following. As the first step, PSAB block receives an incoming packet data from network interface (Gigabit Ethernet links). And then the incoming packet is delivered to logic for searching the predefined patterns. If it is involved in specific rule pattern,

then FF (Fixed Field) flag for interfacing with matching function of kernel logic is set to '1'. Otherwise, FF flag is set to '0'. The flow for preprocessor filtering also begins at the same starting point, and delivers an incoming packet data to logic for checking out the preprocessor filtering rules. If actual preprocessing about an incoming packet data is necessary to be performed, then PP (PreProcessor) flag for interfacing with preprocessing function of kernel logic is set to '1'. Otherwise, PP flag is set to '0'. As a result of these packet processing, if ether PP flag or FF flag is '1', then PSAB block sends the matching packet data to kernel logic. Otherwise, PSAB block receives new incoming packet from network interface, and performs packet processing repeatedly as above. Through the packet processing as this, PSAB block reduces a volume of packets handled by kernel logic to the minimum.

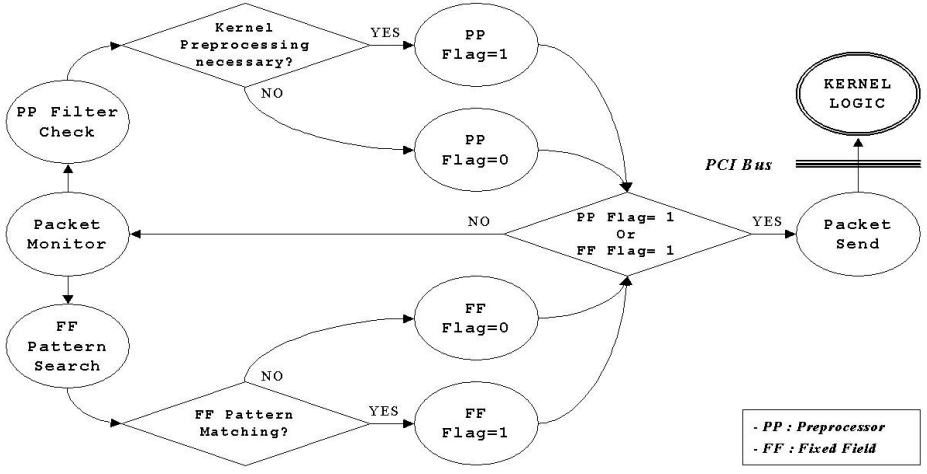


Fig. 3. Detection Algorithm in FPGA Logic

Second, the major functionality of kernel logic is to perform the payload pattern matching and preprocessing about matching packets from PSAB block. And, It mainly is performed by IDAB block in components of SGS. Therefore, detection algorithm of IDAB block is very important as final step for intrusion detection. Most of all, IDAB block finally determines whether received packet is intrusion or not, and sends alert message to CPAB block as a result of analysis. First, function for payload pattern matching is based on direct searching and matching approach about predefined payload patterns. In other words, it seeks to discover network intrusions by testing properties in payload patterns that have identification coincided with matching identification of each fixed field pattern. Second, function for preprocessing decodes the packet payload according to a kind of application services, such as HTTP, Telnet, FTP, and RPC. Besides, it detects the protocol anomaly by performs function, such as ip de-fragmentation, tcp reassembly, and protocol normalization. These functions for preprocessing are run before the payload pattern matching is performed, but it is only performed as occasion demands.

As shown in the figure 4, kernel logic has the serial packet processing flow for preprocessing and payload pattern matching. First, IDAB block receives a matching packet data inspected by PSAB block as the first step for performing its own logic. And then, the matching packet data is decoded to the information required for performing the detection flow of kernel logic. As result of packet decoding, if PP flag is set to '1', then function for preprocessing is run. Otherwise, next step is going on. If result of preprocessing is detected a suspicious packet as the protocol anomaly, then alert message is sent to Application Task block. Otherwise, next step is going on. As the next step, if FF flag is set to '1', then the matching packet data is delivered to logic for searching the payload patterns that have the same matching identification. If it is matched with existing payload patterns, then alert message generated from matching pattern is sent to Application Task block. Otherwise, IDAB block receives new matching packet from PSAB block, and performs packet processing repeatedly as above.

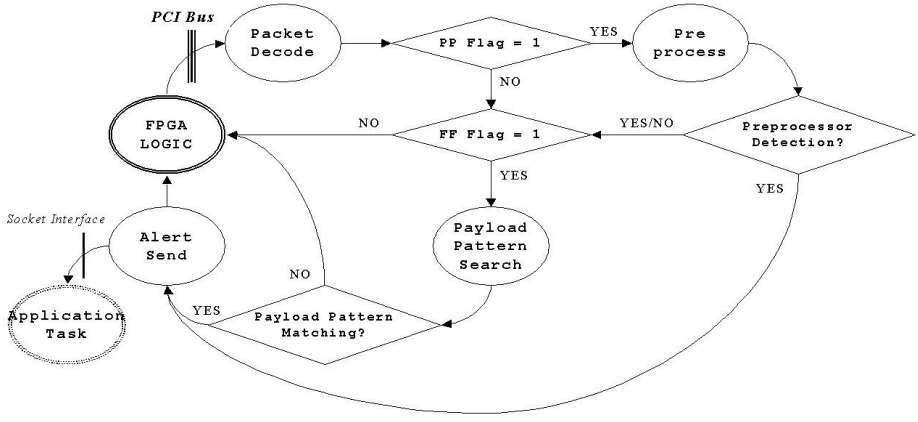


Fig. 4. Detection Algorithm in Kernel Logic

4 Implementation

We have developed our prototype based on the NSCS architecture. The prototype we have developed is programmed in a combination of Java and C, verilog programming language. Most of all, SGS is implemented in programming languages that is best suited for the task it has to perform. Basically, application tasks of SGS are implemented in C programming language, but IDAB block of SGS is implemented to the kernel module programming that is best suited for high-speed pattern matching operation. PSAB module of SGS is implemented in verilog HDL (Hardware Description Language) that is best suited for high-speed packet processing in H/W. Most of all, the prototype we have developed focuses on kernel logic and FPGA logic for real-time traffic analysis and intrusion detection on high-speed links. Also, we employed

inline mode capable of effective response by using two Gigabit Ethernet links as shown in the figure. That is, our prototype has developed in the side of improvement in performance for packet processing.

In our prototype, FPGA logic performs many functions, such as wire-speed forwarding, 5-tuple (protocol, source/destination address, source/destination port) based flow classification, packet sensing, and fixed field pattern matching. Kernel logic also performs many functions, such as preprocessing, payload pattern matching, alert generation, and detection rule management. Besides, mysql database server is employed by SGS as database server for managing security-relevant information and policy information. On the other hand, CPCS manages to its own information by using oracle database server. Some functions such as reporting and communication are common to all SGS and CPCS, and can be provided through shared libraries or similar mechanisms. Finally, for testing of our prototype, CPCS console has implemented in Java2 and HTML for security manager to support comfortable management in Web.

Fig. 5. SGS Security Card Prototype

Currently, we are in the process of improving the implementation as well as developing new ones. That is, our prototype leaves much to be desired. Furthermore, we analyzed the functions of various intrusion detection systems in our testbed network. And now, we are defining more effective analysis functionality in order to improve the performance of detection mechanism on high-speed links.

5 Conclusion and Future Work

In this paper, we designed the architecture of our system, named 'SGS' that performs the real-time traffic analysis and intrusion detection on high-speed links, and proposed the detection mechanism and rule distribution technique that supports more efficient intrusion detection. The detection mechanism is run by the coordination of FPGA logic and kernel logic for improvement in performance. That is, the proposed system focuses on reducing a lowering of performance caused by high-speed traffic analysis to the minimum. Also, it is capable of supporting the effective response by using inline mode monitoring technique on two Gigabit links. According to these

design specifications, we have developed the prototype of our system for the analysis of the traffic carried by Gigabit links. However, the current prototype is very preliminary and a thorough evaluation is required experimentation in a real-world environment. In future, for demonstrating the superiority of the proposed detection mechanism, we will keep up our efforts for improvement in performance of detection mechanism on high-speed links. Finally, we will implement and expand our designed system and give more effort to demonstrate effectiveness of our system.

References

1. Kruegel, C., Valeur, F., Vigna, G. and Kemmerer, R. "Stateful intrusion detection for high-speed networks", In Proceedings of the IEEE Symposium on Security and Privacy, pp. 266-274, 2002.
2. Byoung-Koo Kim, Jong-Su Jang, Sung-Won Sohn and Tai M. Chung, “Design and Implementation of Intrusion Detection System base on Object-Oriented Modeling", In Proceedings of the International Conference on Security and Management, pp. 10-15, June, 2002.
3. H. Debar, M. Dacier and A. Wespi, "Research Report Towards a Taxonomy of Intrusion Detection Systems", Technical Report RZ 3030, IBM Research Division, Zurich Research Laboratory, Jun., 1998.
4. S. Kumar and E. Spafford, "A pattern matching model for misuse intrusion detection", In Proceedings of the 17th National Computer Security Conference, pp. 11-21, Oct., 1994.
5. M. Roesch. "Snort-Lightweight Intrusion Detection for Networks". In Proceedings of the USENIX LISA ’99 Conference, November, 1999.
6. Marcus Ranum, "Burglar Alarms for Detecting Intrusions", NFR Inc., 1999.
7. Thomas Ptacek and Timothy Newsham, "Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection", Secure Networks Inc., 1998.
8. ISS. RealSecure Gigabit Network Sensor. http://www.iss.net/products_services/enterprise_protection/rsnetwork/gigabitsensor.php, September, 2002.
9. Symantec. ManHunt. http://enterprisesecurity.symantec.com /products/products.cfm?ProductID=156, 2002.
10. CISCO. CISCO Intrusion Detection System. Technical Information, November, 2001.
11. W. Richard Stevens, TCP/IP Illustrated Volume I: The Protocols, Addison Wesley, 1994.

An Authenticated Key Agreement Protocol Resistant to a Dictionary Attack

Eun-Kyung Ryu, Kee-Won Kim, and Kee-Young Yoo

Department of Computer Engineering, Kyungpook National University,
Daegu 702-701, South Korea
{ekryu, nirvana}@infosec.knu.ac.kr, yook@knu.ac.kr

Abstract. Recently, Lee-Lee pointed out that Hsu et al.'s key agreement scheme suffers from a modification attack and described an enhancement on it. Both of Lee-Lee's enhancement and Hsu et al. scheme can be considered as variants of Diffie-Hellman scheme with user authentication that are based on a shared-password for providing authentication. This paper shows both schemes cannot withstand to a dictionary attack. Such an attack illustrates that extreme care must be taken when passwords are combined to provide user authentication in cryptographic protocols. This paper also presents a new authenticated key agreement protocol that is not secure to the dictionary attack but also has many desirable security properties, including forward secrecy and known-key secrecy. It is also able to withstand to both passive and active attacks. The security of the proposed scheme is based on the well-known cryptographic assumptions.

1 Introduction

Key agreement protocol is the process in which two communication parties establish a shared secret key using information contributed by both of them. The key may subsequently be used to achieve some cryptographic goal, such as confidentiality or data integrity. Secure authenticated key agreement protocols are important as effective replacements for traditional key establishment achieved using expensive and inefficient couriers[1].

Broadly speaking, key agreement protocols can be classified into two principal techniques, symmetric protocols and asymmetric protocols, by information contributed by legitimate parties that is used to derive a shared secret key. In symmetric protocols the two parties possess common secret information in advance, while in asymmetric protocols the two parties share only public information that has been authenticated.

In particular, many works have focused on symmetric setting in which the parties only use a pre-shared secret, such as password; no supplementary keys or certificates are required. Since the password-based mechanism allows people to choose their own passwords with no assistant device to generate or store, it is the most widely used method for user authentication. This paper considers key agreement protocols for the symmetric setting.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 603–610, 2004.
© Springer-Verlag Berlin Heidelberg 2004

Simple authenticated key algorithm(SAKA) [3] is a symmetric key agreement scheme in which the communication parties use only a pre-shared password. SAKA is considered as an efficient one in terms of computational time and exchanged messages [4]. However, the protocol was found to have security flaws, and then was subsequently improved to eliminate such problems in [4,5,6]. In 2003, Hsu et al [7] proposed a modified authenticated key agreement protocol that is a variant of SAKA. Later, Lee-Lee [8] pointed out that Hsu et al.'s key agreement protocol suffers from a modification attack and described an enhancement of Hsu's scheme to eliminate such a problem.

This paper shows both of Hsu et al's scheme and Lee-Lee's enhancement cannot withstand to a dictionary attack. Such an attack illustrates that extreme care must be taken when passwords are combined to provide user authentication in cryptographic protocols. This paper also presents a new authenticated key agreement protocol that is not secure to the dictionary attack but also has many desirable security properties of forward secrecy and known-key secrecy. It is also able to withstand to both passive and active attacks. The security of the proposed scheme is based on the well-known cryptographic assumptions.

The remainder of this paper is organized as follows. In section 2, we begin by briefly reviewing Hsu et al's protocol and Lee-Lee's enhanced scheme and discuss the security weakness of both schemes. In section 3, we demonstrate a new authenticated key agreement protocol resistant to the dictionary attack. In section 4, we firstly examine desirable security properties for secure authenticated key agreement protocols and then discuss the security of our proposed scheme. Finally, conclusion is given in section 5.

2 Related Works

In this section, we briefly review of Hsu et al's protocol and Lee -Lee's enhancement on it. Then discuss a security flaw of both schemes.

2.1 Hsu et al's Protocol

In their scheme, system parameters are n and g, where n is a large prime and g is a generator with order $n-1$ in GF(n) as the original Diffie-Hellman scheme. All exponentiations are performed modulo n. They assume that two communication parties, called Alice and Bob, have a common pre-shared secret password P. Alice and Bob can pre-compute two integers Q and $Q^{-1} \pmod{n-1}$ from P in any predetermined way before performing the key agreement protocol. To establish a session key, Alice and Bob engage in a protocol run. The protocol consists of two phases: key establishment and key validation phase.

Key establishment phase: Alice(A) and Bob(B), each selects random values a, b in Z_n as their ephemeral keys. They then exchange messages to generate a session key as follows:

Step 1. A $\rightarrow$ B: $X_1 = g^{aQ}$
Step 2. B $\rightarrow$ A: $Y_1 = g^{bQ}$

Then Alice computes the session key $K_1 = Y^a = g^{ab}$, where $Y = Y_1^{Q^{-1}}$. Bob also computes the session key $K_2 = X^a = g^{ab}$, where $X = Y_1^{Q^{-1}}$.

Key validation phase: To check the validity of the established session key, Alice and Bob exchange more two messages as follows:

Step 3. A $\rightarrow$ B: $X_2 = H(id_A, K_1)$
Step 4. B $\rightarrow$ A: $X_2 = H(id_B, K_2)$

After step 3, Bob check if $X_2 = H(id_A, K_2)$ holds or not. If it holds, he authenticates Alice. After step 4, Alice also check if $Y_2 = H(id_B, K_1)$ holds. If it holds, she also authenticates Bob. Above all steps correctly perform, two entities Alice and Bob are convinced the session key $K_1 = K_2 = g^{ab}$

2.2 Lee-Lee's Enhancement

Lee-Lee claimed that Hsu et al.'s protocol can not withstand a modification attack. Consider a following scenario as described in [8]: An adversary, called Eve, can replace X_1 with $X_1' = X_1^t$ and Y_1 with $Y_1' = Y_1^t$ in the step 1 and step 2 of the key establishment phase, and then sends X_1' and Y_1' to Bob and Alice respectively. Finally, Alice and Bob have the same wrong session key $K_1' = K_2' = g^{abt}$. Thus they claimed that Hsu et al's protocol is vulnerable to the modification attack. As an enhancement of Hsu et al' to resist such an attack, they suggested that the key validation phase of Hsu et al's should be modified as following:

Step 3. A $\rightarrow$ B: $X_2 = H(id_A, X_1, K_1)$
Step 4. B $\rightarrow$ A: $Y_2 = H(id_B, Y_1, K_2)$

2.3 Security of Both Schemes

Although Lee-Lee's protocol tried to remove the modification attack on Hsu et al.', both schemes still have an important security flaw. Let us consider the following scenario at both schemes: To establish a session key with Bob, Alice computes $X_1 = g^{aQ}$ and send it to him. An adversary, called Eve(E), who is in a position of Bob receives X_1 from Alice and compute $Y_1' = g^e$ sends to Alice, where e is a random number chosen by Eve. The message flows are as follows:

Step 1. A $\rightarrow$ E: $X_1 = g^{aQ}$
Step 2. E $\rightarrow$ A: $Y_1' = g^e$

Then Alice compute the session key $K_1 = (Y_1'^{Q^{-1}})^a = g^{aeQ^{-1}}$. To verify the established session key, Alice compute X_2 and then sends X_2 to Eve as following:

Step 3. A $\rightarrow$ E: $X_2 = H(id_A, K_1)$ or $X_2 = H(id_A, X_1, K_1)$

Note that after this step 3 Eve can successfully obtain the shared-password P by performing a dictionary attack or an off-line password guessing attack. For the dictionary attack, Eve can take a list of probable passwords and derive the corresponding Q' and $Q'^{-1} \pmod{n-1}$ for all in the list. For each Q'^{-1}, Eve then computes $K_2' = X_1^{e(Q'^{-1})^2}$, and $Y_2' = H(id_A, K_2')$ or $Y_2' = H(id_A, X_1, K_2')$, depending on the protocol. Then check if there is any Q'^{-1} that holds $Y_2' = X_2$. If $Y_2' = X_2$, the attack succeeds. In a case of the off-line password guessing attack, we could consider it in a similar way. First, Eve can make a guess at the shared-password P and derive Q' and Q'^{-1} from the guessed password P'. Then compute $K_2' = X_1^{e(Q'^{-1})^2}$, and $Y_2' = H(id_A, K_2')$ or $Y_2' = H(id_A, X_1, K_2')$ with the same way as above. Then check if $Y_2' = X_2$. If $Y_2' = X_2$, the attack succeeds. If not, the adversary repeatedly performs it until it holds.

Unlike typical private keys, the password has limited entropy, constrained by the memory of the user. Roughly speaking, the entropy of human memorable password is about 2 bits per character. Therefore, the goal of obtaining a legitimate communication parties' shared-password by the adversary can be achieved within a reasonable time. Thus, the dictionary attack or the password guessing attack for both schemes should be considered as a realistic one.

Table 1. Notation

A, B	two communication parties, Alice and Bob
p	A large prime
q	A prime with $q = (p-1)/2$
g	A generator with order q in Z_p
π	The user's password
a, b	Ephemeral private keys of A and B
$H()$	One-way hash function
S_{AB}	The shared secret calculated by the principals
K_{AB}	The derived session key

3 The Proposed Scheme

In this section we describe a new authenticated key agreement protocol resistant to the dictionary attack.

3.1 System Setup

We assume that two communication parties, called Alice and Bob have a shared-password π in advance. In our protocol, all computations are performed in a finite field GF(p). The system parameters p, q and g are well-known values, agreed to beforehand. Table 1 shows the notation used in our protocol.

3.2 Protocol Run

To establish a session key, Alice and Bob engage in an instance of the protocol run. Each, Alice(A) and Bob(B), selects random values a, b in Z_q as their ephemeral keys. The message flows of the protocol run are as follows:

Step 1. A $\rightarrow$ B: $X = g^a + h(\pi)$
Step 2. B $\rightarrow$ A: $Y = g^b, V_B = H(id_A, X, S_{BA})$
Step 3. A $\rightarrow$ B: $V_A = H(id_B, Y, S_{AB})$

In step 1, Alice computes $X = g^a + h(\pi)$ and sends it to Bob. In step 2, Bob computes $Y = g^b$ and $V_B = H(id_A, X, S_{BA})$, where $S_{BA} = (X - H(\pi))^b = g^{ab}$. Then send them back to Alice. Alice checks if $V_B = H(id_A, X, S_{AB})$, where $S_{AB} = Y^a = g^{ab}$. If it is correct, Alice computes $V_A = H(id_B, Y, S_{AB})$ and sends it to Bob, Finally, Bob checks if $V_A = H(id_B, Y, S_{BA})$. If it is correct, Bob also authenticates Alice.

The session key of both parties is then $K_{AB} = kdf(id_A, id_B, S_{AB}) = kdf(id_A, id_B, S_{BA})$, where kdf is a key derivation function. Once the protocol run completes successfully, both parties may use K_{AB} to encrypt subsequent session traffic.

4 Security Analysis

In this section, we firstly examine some desirable security properties required for secure authenticated key agreement protocols. Then, discuss the security of our proposed scheme that is based on the well-known cryptographic assumptions such as the Diffie-Hellman(DH), the one-way hash(OWH) and Discrete Logarithm(DL) assumptions.

As described in [1] a secure key agreement protocol should be able to withstand both *passive* attacks(where an attacker attempts to prevent a protocol from achieving its goals by merely observing honest entities carrying out the protocol) and *active* attacks(where an attacker additionally subverts the communications by injecting, deleting, altering or replaying messages). In addition, the following *security properties* should be considered since they are often desirable, depending on application domain. In the following, A and B are honest parties.

- *known-key security*. Each run of a key agreement protocol between two entities A and B should produce a unique secret key; such keys are called

session keys. A protocol should still achieve its goal in the face of an attacker who has learned some other session keys.

– *forward secrecy.* If long-term private keys of one or more entities are compromised, the secrecy of previous session keys established by honest entities is not affected.

Lemma 1 (The OWH assumption). *It is infeasible to find x such that $H(x) = y$ for a given y, and it is infeasible to find a pair (x, x') such that $x \neq x'$ and $H(x) = H(x')$.*

Lemma 2 (The DL assumption). *It is hard to find the integer x, $0 \leq x \leq p-2$, such that $g^x \equiv y \,(mod\, p)$, given a prime p, a generator g of $y \in Z_p^*$, and an element $y \in Z_p^*$.*

Theorem 1 (The passive attack). *An adversary who eavesdrops on a successful protocol run cannot make a guess at the session key using only information obtainable over network and a guessed value of the password π.*

Proof. To compute session key, the adversary should derive a or b from the publicly-visible information X, Y, V_A, and V_B. Under OWH assumption, the message V_A and V_B leak no information to the adversary without the ability to compute keying material g^{ab}. Consequently, the adversary should compute a from X or b from Y. However, by lemma 2, the a and b are protected under the DL assumption.

Theorem 2 (The impersonation attack). *An adversary cannot impersonate legitimate communication parties without the knowledge of shared password π.*

Proof. To impersonate one of legitimate communication parties, the adversary should derive the correct verification message V_A or V_B. However, the problem is combined with solving the discrete logarithm and making a good guess at the password π. Suppose that an adversary wants to fool Bob into thinking he is talking to Alice. First, she can compute $X' = g^e + h(\pi')$ and send it to Bob. Bob computes $Y = g^b$ and $V_B = h(id_A, X', K_{BA'})$ and sends them to the adversary. When the adversary receives Y and V_B from Bob, she has to compute $V_E = h(id_B, Y, K_{A'B})$ where $K_{A'B} = (X' - h(\pi))^b$ and sends it to Bob. Consequently, the adversary should compute b from Y and make a correct guess at the shared password π. However, by lemma 2, the a is protected under the DL assumption.

Theorem 3 (The known-key attack). *An adversary with information about a past session key from an eavesdropped session cannot derive and use it either to gain the ability to impersonate the user directly or to conduct a brute-force search against the user's password.*

Proof. If a past session key is revealed to an adversary, the adversary does not learn any new information from combining the past session key with the publicly-visible information X, Y, V_A or V_B. This is true because the message V_A or V_B leak no information to the adversary under OWH assumption. In theorem 1, we have already established that the adversary cannot make meaningful guesses at the session key K from guessed passwords, and there does not appear to be any easier way for her to carry out a brute-force attack.

Lemma 3 (The DH assumption). *Given a prime p, a generator g and elements* $g^a \ (mod\, p)$ *and* $g^b \ (mod\, p)$*, it is infeasible to compute* g^{ab}

Theorem 4 (The forward secrecy). *It does not allow the adversary to determine the session key for past sessions and decrypt them even if the user's password itself is compromised,*

Proof. To compute a past session key, the adversary should derive the keying material g^{ab} from g^a and g^b. However, it is clearly an instance of DH problem. Therefore, it preserves the forward secrecy by lemma 3.

5 Conclusion

In this paper, we showed that both of Hsu et al.'s protocol and Lee-Lee' enhancement are vulnerable to a dictionary attack. We also presented a new authenticated key agreement protocol that does not eliminate such a security flaw but also has many desirable security properties, including forward secrecy and known-key secrecy. It is also able to withstand to both passive and active attacks.The security of our proposed protocol is based on the well-known cryptographic assumptions: the Diffie-Hellman(DH), the one-way hash(OWH) and Discrete Logarithm(DL) assumptions.

Acknowledgments. We would like to thank anonymous reviewers for the helpful comments. This work was supported by the Brain Korea 21 Project in 2003.

References

1. S. Blake-Wilson, A. Menezes: Authenticated Diffie-Hellman key agreement protocols. Proceedings of the 5th Annual Workshop on Selected Areas in Cryptography (SAC '98), Lecture Notes in Computer Science 1556. 1999, pp. 339-361
2. W. Diffie, M. Hellman, New directions in cryptography, IEEE Transaction on Information Theory, IT-22, 1976, pp. 644-654
3. D.H. Seo, P. Sweeney: Simple authenticated key agreement algorithm. Electronics Letters. June 1999, Vol. 35, pp. 1073-1074
4. Y.M. Tseng: Weakness in simple authenticated key agreement protocol. Electronics Letters. Jan. 2000, Vol. 36, pp. 48-49

5. I.C. Lin, C.C Chang, M.S. Hwang: Security enhancement for the simple authentication key agreement algorithm. Computer Software and Applications Conference(COMPSAC) 2000, pp. 113-115
6. W.C. Ku, S.D. Wang: Cryptanalysis of modified authenticated key agreement protocol. Electronics Letters. Oct. 2000, Vol. 36, pp. 1770-1771
7. C.L. Hsu, T.S. Wu, T.C. Wu, C. Mitchell: Improvement of modified authenticated key agreement protocol, Applied Mathematics and Computation 142. 2003, pp. 305-308.
8. N.Y. Lee and M.F. Lee: Further improvement on the modified authenticated key agreement scheme, Applied Mathematics and Computation, Available online Nov. 2003.

A Study on Marking Bit Size for Path Identification Method: Deploying the Pi Filter at the End Host*

Soon-Dong Kim[1], Man-Pyo Hong[1], and Dong-Kyoo Kim[2]

[1]Graduate School of Information Communication, Ajou University, Suwon, Korea.
{sdkim, mphong}@ajou.ac.kr
[2]College of Information Technology, Ajou University, Suwon, Korea.
{dkkim}@ajou.ac.kr

Abstract. Recently, DDoS attacks are more and more serious to the Internet. Many specialists research the defending methods against DDoS. Pi had been proposed as one of the defense methods against complicated DDoS attack by spoofed IP address. Pi is a new packet marking approach, and Pi enables a victim to identify packets traversing the same paths through the Internet on a per packet basis, regardless of source IP address spoofing. Marking size of Pi is the most important parameter of Pi marking scheme to decide the performance of Pi. At the end hosts' view, the most proper marking size of Pi is affected by the Internet environment and its topology. In existing Pi scheme, Pi filter deployed on the ISP's side of the last hop link, but this paper consider the Pi filter deployed at end host in the ISP and tried to find the most proper marking size.

1 Introduction

Recently, DDoS attacks are more and more serious to the Internet. In a typical DDoS attack, attackers deploy multiple agents and use them to attack by sending large packets. DDoS attacks had shut down several large Internet site, such as Yahoo! and eBay. Recently, on Saturday, 25, January, 2003, Sapphire/Slammer Worm plagued the Internet [4]. As larger as the Internet size, the threats of the DDoS would be serious. IP protocol's flaw that attacker can spoof IP address of the attack packet make it difficult to identify and block the packet with the packet's source IP address under current Internet infrastructure. Because of this, many researchers have studied and proposed common countermeasure is IP traceback [3]. However this mechanism have a few shortcomings such that victim must make a collection of specific amount of packet for reconstruct the exact path [1]. Not only attackers can use this property to server resource attack, but also the processes take time and this could induce DoS. One of the other countermeasures against DDoS is Pi [1]. Routers that are deploying Pi module mark hashed value of the routers' IP addresses with marking size n on the identification field of incoming packets. These packets to the victim could have a marked identification field of packets as an identifier based on the router path that the packets traverse. After learning phase, the victim has attack markings list, and the

* This study was supported by the Brain Korea 21 Project in 2004.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 611–616, 2004.
© Springer-Verlag Berlin Heidelberg 2004

victim filter the packet if incoming packet's identification field is one of the attack markings list.

On Pi scheme, marking size n is the most important parameter. Performance of Pi is sensitive to the values of marking size n=1, 2, and so on. We tried to find the most proper n value with comparison of each value of n.

The remainder of this paper is organized as follow: In section 2, we present some assumptions that are needed to propose our study in this paper. In section 3, we explain the motivation of this consideration about Pi marking size. In section 4, we simulate the experimental performance. In section 5, we show the result. Finally, in section 6, we conclude the paper.

2 Assumptions

We assume that Pi filter is deployed at the end host on an AS (Autonomous System). This assumption is one kind of countermeasure of server resource attack. The server resource attacks are divided to the following two types.

- Server Processing Attack: In this attack, for example, the attackers send many useless packets. Therefore if the amounts of packets to process exceed victim's processing ability, then some packets are dropped without processing.
- Server Memory attack: In the case of TCP_SYN flooding attacks [5], attackers take the flaw in protocols to deplete the memory of a server.

A victim can filter incoming packets by matching the packets' identification field with attack markings list of the victim. Therefore Pi has some advantages against these attacks because of packet by packet basis filtering. Finally, we assume that all routers of the Internet support Pi marking scheme.

3 Motivation of the Size of Marking Bit

3.1 AS Topology

AS has some topologies, tree, mash, and etc. However the end hosts's point of view, except intra networking, only a few paths are used to communicate other host in another AS in the Internet. It seems tree topologies. According to Skitter Map's traceroute data [2], as shown in Table 1, upstream paths are approximately same from the end host to the third router. This presents to us that, in many case, the marked data of third hop (or second hops) from the end host are meaningless to verify the path of a packet because the data may be always same. For simple test, we assume that the paths are always same from the end host (later, we call this host to 'victim') to the third upstream router to communicate between each AS.

According to the assumption, 1 bit and 2 bit markings variances are like following.

1-bit marking: Exp(2, 16-3) = 8182
2-bit marking: Exp(2, 16-6) = 1024

Table 1. The most frequently used IP address's frequency to communicate with other host

Data Sets	1st hop	2nd hop	3rd hop	comments
Data1	4131	4131	4130	Single path only
Data2	4042	3370	1636	Two path at 3rd router
Data3	4873	4872	4337	Almost single path
Data4	4286	1344	1156	Well distributed
Data5	4806	1139	631	Well distributed
Data6	4951	2481	2481	Almost two path from 2nd router
Data7	3374	3374	3363	Almost single path

Because of the three hops that are always same, 3 bits in incoming packets identification field are wasted when 1-bit marking scheme. According to the same reason, 6 bits are wasted when 2-bit marking scheme.

Pi values that traverse route a path less than 16 hops (or 8 hops when n=2) could have meaningless unmarking bits. For example, when hop count[1] is 13, the effective marking bit and the variance are following.

$$\text{1-bit marking: } Exp(2, 13-3) = 1024$$
$$\text{2-bit marking: } Exp(2, 16-6) = 1024$$

If hop count is greater than 13, Pi of 1-bit marking scheme is more effective because of more variance approximately Exp(2, (hop count-13)) than 2-bit marking scheme, however if hop count is less than 13, Pi of 2-bit marking scheme is more effective approximately EXP(2, hop count) than 1-bit marking scheme.

3.2 Internet Path Length and Marking Size

Pi's performance is sensitive to hop count, and total numbers of marked hops affect false positive rate or false negative rate. For example, if hop count is 10, then 1-bit marking scheme has 6 bit garbage value. These values cause false negative. In section 4.3 we show the detail about this. Additionally we see [1], most of the Internet paths are concentrated on around 16 hops count. This is very important data because it is obvious that the scheme is better which can work better at the most cases.

The previous paper [1] has detail consideration of marking bit size. The paper presented that only 1-bit and 2-bit marking schemes are considerable in Internet environment. 1-bit marking scheme can mark 16 routers' information. However this scheme uses a bit to present a router's information, 2-bit marking scheme has more distinction than this one.

4 Experimental Performance

In this section, we evaluate Pi's performance under DDoS attack. We are going to define sample Internet data set and DDoS attack model. We will explain the

[1] Hop count: We say the hop count to the number of routers that a packet traverses from source to destination. This value plus one and generally called 'hop count' are same.

experiment design and performance metrics, and finally we present the results of our experiments.

4.1 Internet Data Sets

In our experiments, we use the Internet topology: CAIDA's Skitter Map [2]. This topology was created by using a single host send traceroutes to the other hosts through out Internet and recording the paths as the IP address of the routers along each route. In our experiments, we used only the complete records of the total records for more reliable results. Additionally, we define the source address of the records as victim and the end hosts on the traceroute paths as our legitimate users or attackers.

4.2 DDoS Attack Model

In order to filter attack packets, the victim must have some ways to differentiate them from normal packets. For that, the victim has attack markings list. If the victim concludes a packet to be attack, the packet's identification field value is added to attack markings list. We model our DDoS attack in two phases- learning phase and attack phase [1]. We simplify these two phases for simple experiment. To define an algorithm for attack packet identification is outside scope of this paper. In the first phase, the learning phase, we select some attackers and send a packet to victim. The victim records the markings to the attack markings list. After the learning phase, the attack phase, the victim decides whether the victim drops the packet by comparing marking of incoming packet with the markings in the attack markings list.

4.3 Experiment Design, Attack Scenario, and Performance Metric

First, we choose approximately 207,000 paths randomly from CAIDA's Skitter Map as normal hosts. We also choose some attackers in the normal hosts. If one host is selected as an attacker, we assume that the packets to be sent from the attacker are always attack packets. All identification fields of packets on this experiment are filled with random garbage value. The packets through a routing path, each routers mark router's marking value on identification field of the packet. If an end host is in 8 hops counts distance, in case of 1-bit marking scheme, at least 8 bits are filled with garbage value when the victim receives the packet. At the learning phase, as we said above, a victim can make 16 bits attack markings list, but it is difficult to determine whether the bits in an attack marking are garbage or not. If 8 bits of the marking are garbage, many false negatives[2] could occur. On the other hand, in case of 2-bit marking, if attacker's distances are more than 8 hops count, near routers to the victim overwrite the previous routers markings. Therefore there is a strong possibility that some hosts have same Pi marking values. These cause the false positive[3] if the marking values are in the attack markings list of victim. Therefore we define as performance metric to the

[2] False negative: If attacker sends attack packets, but the victim cannot detect the attack.

[3] False positive: If normal user sends normal packets, but the filter of the victim catches the packet to the attack packets.

false rate. For the measurement of performance, we define the false rate-the sum of false negative rate and the false positive rate.

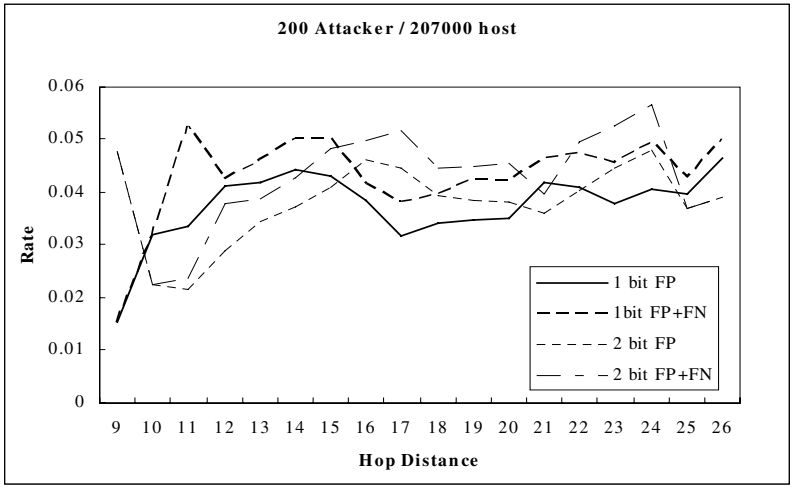


Fig. 1. Pi filter performance; the false positive and negative rates are calculated at each hop distances

5 Result

In Fig. 1, we could see the false rate in n=1 and n=2 bit schemes divided by each hop counts. We test approximately 207,000 distinct paths and set the 200 number of attack nodes and generate one million packets randomly. Fig. 1 presents the result at hop distances from 9 to 26. The experiment's results at hop distances from 1 to 8 are not enough to test.

As shown in Fig. 1, most of the false rates defend on false positive rate. Relatively false negative rates are low because most attacks or normal communications are enough hop distances to fully mark the identification field of IP packet.

2-bit marking scheme's false positive rates of the domain that its hop counts, less than 13, are lower than 1-bit marking scheme. As we can see at Fig. 1, the 1-bit curve has less false rate in the domain from 13 to 20 hops count. Less than 13 hops distances, 2-bit marking scheme's false rates are lower than 1-bit marking scheme. These results well match to our previous forecast at section 3.1.

6 Conclusion

In this paper, we have considered Pi marking size, n=1 or n=2. Each scheme has some advantages and disadvantages. We measured the efficiency of these schemes by false rate. It is sum of the false positive rate and the false negative rate. 2-bit scheme has

better performance in relatively short hops count domain approximately with less than 13 hops count distance. On the other hand, 1-bit scheme performs more accurately in the domain that has approximately more than 13 hops count distance. At the end host's point of view, therefore 1-bit scheme is proper than 2-bit one because most of the Internet hosts communicate with each other via longer distances than 13 hops count.

References

1. Yaar, A., Perrig, A., Song, D.: Pi: A Path Identification Mechanism to Defend against DDoS Attacks. Proceeding of Symposium on Security and Privacy 2003. (2003) 93–107
2. CAIDA. Skitter. http://www.caida.org/tools/measurement/skitter/ (2000)
3. Chen, Z., Lee, M.: An IP traceback technique against denial-of-service attacks. Proceeding of 19th Annual Computer Security Applications Conference (2003) 96–104
4. Berkeley University. The Spread of the Sapphire/Slammer Worm http://www.cs.berkeley.edu/~nweaver/sapphire/ (2002)
5. Computer Emergency Response Team(CERT). TCP_SYN flooding and IP spoofing attacks. Technical Report CA-96:21. Carnegie Mellon University. Pittsburgh, PA (1996)

Efficient Password-Based Authenticated Key Agreement Protocol

Sung-Woon Lee[1], Woo-Hun Kim[1], Hyun-Sung Kim[2], and Kee-Young Yoo[1]*

[1] Department of Computer Engineering Kyungpook National University, Taegu, KOREA,
{staroun, whkim}@infosec.knu.ac.kr, yook@knu.ac.kr
[2] Department of Computer Engineering Kyungil University, Kyungsansi, Kyungsangpookdo, KOREA
kim@kiu.ac.kr

Abstract. In this paper, we present a new password-based authenticated key agreement protocol called PAKA, which provides mutual authentication and key agreement over an insecure channel between two parties knowing only a small password having low entropy. We then extend PAKA to a protocol called PAKA-X, in which the client uses a plaintext version of the password, while the server stores a verifier for the password, and which does not allow an adversary who compromises the server to impersonate a client without actually running a dictionary attack on the password file. The proposed protocols are secure against passive and active attacks and provide perfect forward secrecy.

1 Introduction

It is necessary to verify the identities of the communicating parties when they initiate a connection. This authentication is usually provided in combination with a key agreement protocol between the parties. Techniques for user authentication are broadly based on one or more of the following categories: (1) what a user knows, (2) what a user is, or (3) what a user has. Among them, the first category is the most widely used method due to the advantages of simplicity, convenience, adaptability, mobility, and less hardware requirement. It requires users only to remember their knowledge like a password. However, traditional password-based protocols are susceptible to off-line password guessing attacks (called dictionary attacks) since many users tend to choose memorable passwords of relatively low entropy. Since Bellovin and Merrit [1] presented a protocol called EKE for password-based authentication and key agreement which was resistant to these types of off-line dictionary attacks, many password authenticated key agreement protocols have been proposed.

* Corresponding author: Kee-Young Yoo (yook@knu.ac.kr)

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 617–626, 2004.
© Springer-Verlag Berlin Heidelberg 2004

The following classes of password-authenticated key agreement protocols are defined in IEEE Std1363a-2002 [2].

- Balanced Password-authenticated Key Agreement Schemes [2, 3, 4, 5, 6, 7], in which two parties use a shared password to negotiate one or more shared ephemeral keys such that the shared keys are established if and only if they use the same password.
- Augmented Password-authenticated Key Agreement Schemes (usually called verifier-based protocol) [3, 6, 7, 8, 9, 10, 11, 12], in which two parties (denoted Client and Server) use related password-based values to negotiate one or more shared ephemeral keys such that the shared keys are established if and only if they use values that correspond to the same password. Server uses password verification data (usually called verifier) that is derived from client's password data. The scheme forces an attacker who steals the password verification data to further perform a successful brute-force attack in order to masquerade as client.

In this paper, we present a new password-based authenticated key agreement protocol called PAKA, which provides mutual authentication and key agreement over an insecure channel between two parties knowing only a small password having low entropy. PAKA is secure against passive and active attacks and provides perfect forward secrecy. We then present an extension of PAKA called PAKA-X in which the client stores a plaintext version of the password, while the server stores a verifier for the password. PAKA-X provides added security in the case of server compromise. That means an attacker not being able to pose as a client after compromising the server, however, it would be trivial to pose as the server. Compared to the previously well-known protocols in terms of several aspects including the number of protocol steps, random numbers, exponentiations, hash functions, and symmetric encryption/decryption, our protocols are very simple and efficient.

The remainder of this paper is organized as follows. In section 2, we describe PAKA and PAKA-X protocol. In section 3, we show security analysis of our protocols. In section 4, we compare them to the related protocols. Finally, section 5 gives our conclusions.

2 The Proposed Protocols

2.1 Notations

The following notations are used throughout this paper.

Table 1. Notations for the proposed protocols

Notation	Description
A	Alice's public identity
B	Bob's public identity
π	A weak password of Alice

a, b	Session-independent random numbers
p	A large prime
g	A generator g in the cyclic group Z_p^*
$H(\cdot)$	A collision-resistant one-way hash function
$\oplus$	Bit-wise exclusive-OR (XOR) operation
K	A session key
v	Verifier computed from a password
c^{-1}	Inverse of c on Z_p^*

2.2 PAKA

In this section, we present a new Password-based Authenticated Key Agreement protocol called PAKA, which provides mutual authentication and key agreement over an insecure channel between two parties knowing only a small password having low entropy.

Assume that two communication parties, called *Alice* and *Bob*, share a common memorable password π. *Alice* and *Bob* can pre-compute two integers Q and Q^{-1} from π in any predetermined way before the protocol begins. $h(\cdot) : \{0,1\}^* \rightarrow \{0,1\}^{l(k)}$, where k is our security parameter long enough to prevent brute-forcing attack and $l(k) >= 2k$, means a collision-free one-way hash function, which is assumed to behave like random oracles [13]. We will omit 'mod p' from expressions for simplicity. The steps for the PAKA protocol are as follows:

1. *Alice* computes $X_A = g^{aQ} \oplus Q$ by choosing $a \in_R Z_p^*$, and then sends her *id* and X_A to *Bob*.
2. *Bob* computes $X_B = g^{bQ^{-1}} \oplus Q^{-1}$ by choosing $b \in_R Z_p^*$, and then sends X_B to *Alice*. While waiting for a message from *Alice*, he computes $K_B = (X_A \oplus Q)^{bQ^{-1}}$, $V_A' = h(A, X_B, K_B)$, and $V_B = h(B, X_A, K_B)$ in sequence.
3. After receiving the message from *Bob*, *Alice* computes $K_A = (X_B \oplus Q^{-1})^{aQ} = g^{ab}$ and $V_A = h(A, X_B, K_A)$, and then sends V_A to *Bob*. While waiting for a message from *Bob*, she computes $V_B' = h(B, X_A, K_A)$.
4. After receiving the message from *Alice*, *Bob* checks whether $V_A \stackrel{?}{=} V_A'$ holds or not. If it holds, he is convinced that K_A is validated, and then sends V_B to *Alice*.
5. After receiving the message from *Bob*, *Alice* checks whether $V_B \stackrel{?}{=} V_B'$ holds or not. If it holds, she is convinced that K_B is validated.
6. Finally, *Alice* and *Bob* compute the common session key $K = h(K_A) = h(K_B) = h(g^{ab})$, respectively.

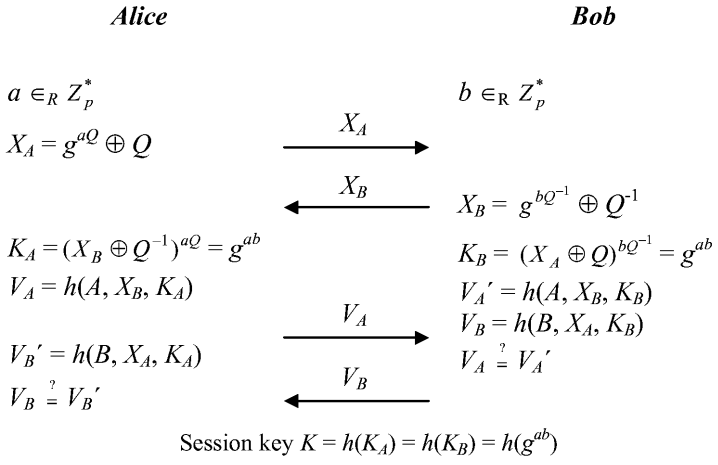


Fig. 1. PAKA protocol

2.3 PAKA-X

In this section, we present an extended PAKA which is secure against server compromise, i.e., an attacker who steals the password file from the server can not use that information directly to impersonate the client. We call this extended protocol PAKA-X (PAKA-eXtended).

In PAKA-X, to be resistant to server compromise, the server does not store the plaintext password. In stead, the server stores a verifier to verify a client's password. The verifier v is the information computed from a password π. We assume there is an initialization in which the client, *Alice*, chooses a memorable password π, computes a verifier $v = g^{h(A,B,\pi)}$, and then sends v to the server, *Bob*, over a secure channel. *Bob* stores (*id*, v), where *id* indicates an identifier or name of *Alice*. To enhance the efficiency of the protocol, $v = g^{h(A,B,\pi)}$ and $h(A,B,\pi)^{-1}$ can be pre-computed by *Alice* before the protocol runs. The steps for the PAKA-X protocol are as follows:

1. *Alice* computes $X_A = g^a \oplus v$ by choosing $a \in_R Z_p^*$ and then sends her *id* and X_A to *Bob*.
2. After receiving the message from *Alice*, *Bob* retrieves v from a password file, computes $X_B = (v)^b \oplus v$ by choosing $b \in_R Z_p^*$, and then sends X_B to *Alice*. While waiting for a message from *Alice*, he computes $K_B = (X_A \oplus v)^b = g^{ab}$, $V_A' = h(A, X_B, K_B)$, and $V_B = h(B, X_A, K_B)$ in sequence.
3. After receiving the message from *Bob*, *Alice* computes $K_A = (X_B \oplus v)^{a \cdot h(A,B,\pi)^{-1}} = g^{ab}$ and $V_A = h(A, X_B, K_A)$, and then sends V_A to *Bob*. While waiting for a message from *Bob*, she computes $V_B' = h(B, X_A, K_A)$.

4. After receiving the message from *Alice*, *Bob* checks whether $V_A \stackrel{?}{=} V_A'$ holds or not. If it holds, *Bob* is convinced that K_A is validated, and then sends V_B to *Alice*.
5. After receiving the message from *Bob*, *Alice* checks whether $V_B \stackrel{?}{=} V_B'$ holds or not. If it holds, *Alice* is convinced that K_B is validated.
6. Finally, *Alice* and *Bob* compute the common session key $K = h(K_A) = h(K_B) = h(g^{ab})$, respectively.

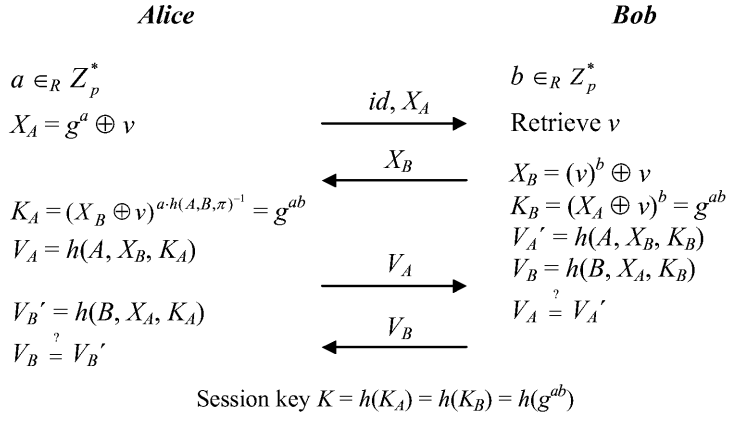


Fig. 2. PAKA-X protocol

3 Security Analysis

In this section, we describe the definitions for the secure protocol and analyze the security of our protocols with regard to several attacks.

For considering the security of our protocol, we assume that the security parameter k has the length long enough to prevent the brute-force attack on itself. Therefore, we could define the negligible probability as follows:

Definition 1. The probability is negligible if it is equal to or less than 2^{-k}.

We use the term of negligible probability for proving the security of our protocol. Suppose that all communication among interacting parties is under the adversary's control as in [13]. In particular, the adversary can read the messages produced by the parties, provide messages of her own to them, modify messages before they reach their destination, delay messages or replay them, and make new instances of any parties. In this paper, the adversary is called *Eve*. Note that the success of *Eve* means she has been accepted. That is, *Eve* must have found a session key or a password. The security of our protocols is based on the difficulty of the discrete logarithm problem and the Diffie-Hellman problem which are believed infeasible to solve in polynomial time. They can be defined as follows:

Definition 2. The computational Discrete Logarithm Problem (DLP) shows that computing a giving g and g^a is difficult problem. We define that the probability of solving the DLP is negligible, i.e., $Pr <= 2^{-k}$.

Definition 3. The computational Diffie-Hellman Problem (DHP) shows that computing gab giving g^a and g^b is difficult problem. We define that the probability of solving the DHP is negligible, i.e., $Pr <= 2^{-k}$.

We now prove the security of PAKA using these definitions.

Theorem 1. PAKA is secure against passive and active attacks.
Proof. We assume that *Eve* succeeds if she find the password π or the session key K. Therefore, we show that probability to succeed in finding them is negligible due to the difficulty of solving DLP and DHP.

1. A completeness of the protocol is already proved by describing the run of the protocol in section 2.
2. The acceptance by both parties means that both V_A and V_B are successfully verified. That is, $h(A, X_B, K_A) = h(A, X_B, K_B)$ and $h(B, X_A, K_B) = h(B, X_A, K_A)$. We show that if it is the case that both parties accept and agree on the same session key, then the probability that *Eve* have modified the messages being transmitted is negligible. The probability of guessing $g^{ab}(=K_A=K_B)$ directly is equal to or less then 2^{-k} . And the only way for *Eve* to find K_A or K_B is to solve DLP and DHP. Therefore, this case is negligible by definition 2 and 3.
3. If *Eve* is benign (a passive attacker), all she can capture are as follows: $X_A = g^{aQ} \oplus Q$, $X_B = g^{bQ^{-1}} \oplus Q^{-1}$, $V_A = h(A, X_B, K_A) = h(A, g^{bQ^{-1}} \oplus Q^{-1}, g^{ab})$, and $V_B = h(B, X_A, K_B) = h(B, g^{aQ} \oplus Q, g^{ab})$. However, it is negligible due to the difficulty of solving DLP and DHP to find K_A or K_B $(= g^{ab})$ from them. And off-line password guessing attack succeeds when there are pieces of information in communications that can be used to verify the correctness of the guessed password. *Eve* first guesses a password π', compute Q' and Q'^{-1} from π' and then tries to verify her guess by using X_A, X_B, V_A, and V_B. However, she has no way verifying her guess from them without solving DLP and DHP.
4. Now, we consider the active adversary and divide it into three cases.
   - Since a and b are selected in the cyclic group for a uniform distribution, we can see that $X_A = g^{aQ} \oplus Q$ and $X_B = g^{bQ^{-1}} \oplus Q^{-1}$ remain on the cyclic group under uniform distribution. There is no way to find the relationship between the rejected password and the remaining password.
   - If *Eve* masquerades *Alice*, she may know a, g^a made by herself and $g^{bQ^{-1}} \oplus Q^{-1}$ sent from *Bob*. It is helpless since there is not any verifiable data for the guessed password. That means she cannot carry out the off-line guessing attacks. Finally, she has to reply with V_A, but the probability of correct answer is negligible.
   - If *Eve* masquerades *Bob*, she may know $g^{aQ} \oplus Q$, $(g^b)^{(g^b \oplus Q^{-1})^{aQ}}$ sent from *Alice* and b, g^b by herself. It is helpless since there is not any verifiable data for the

guessed password. That means she cannot carry out the off-line guessing attacks. Of course, she also cannot reply with valid V_B.

Therefore, we could say our protocol is secure against passive and active attacks. •

Theorem 2. PAKA provides the property of the perfect forward secrecy.
Proof. Perfect forward secrecy is provided in the situation that even though a password is compromised, *Eve* cannot derive previous session keys. To analyze this, suppose that *Eve* knows the password π. Then she tries to find previous session keys from the information collected by passive attack in past communication sessions, i.e., $g^{aQ} \oplus Q$, $g^{bQ^{-1}} \oplus Q^{-1}$, $h(A, g^{bQ^{-1}} \oplus Q^{-1}, g^{ab})$, $h(B, g^{aQ} \oplus Q, g^{ab})$. However, she cannot do these using them without solving DLP and DHP. Therefore, PAKA provides the property of perfect forward secrecy. •

Theorem 3. PAKA is secure against the Denning-Sacco attack.
Proof. To be secure against the Denning-Sacco attack, the protocol should be designed such that even though a session key is compromised, *Eve* cannot compute the password and confirm the correctness of the guessed password. To analyze this, suppose that *Eve* knows a session key $h(g^{ab})$. Then she tries to compute the password or confirm the correctness of the guessed password from it and the information collected by passive attack in past communication sessions, i.e., $h(g^{ab})$, $g^{aQ} \oplus Q$, $g^{bQ^{-1}} \oplus Q^{-1}$, $h(A, g^{bQ^{-1}} \oplus Q^{-1}, g^{ab})$, $h(B, g^{aQ} \oplus Q, g^{ab})$. However, she cannot do these using them without solving DLP and DHP. Therefore, PAKA is secure against the Denning-Sacco attack. •

Similarly, PPAK-X is also secure against passive and active attacks, provides perfect forward secrecy, and is secure against the Denning-Sacco attack. We will omit the security analysis for them and only analyze the security against server compromise.

Theorem 4. PAKA-X is secure against server compromise.
Proof. The protocol being secure against server compromise means an attacker not being able to pose as a client after compromising the server. In PAKA-X, if *Eve* gains password file, she may knows a client's verifier $v = g^{h(A,B,\pi)}$. However, she cannot pose as the client because of not knowing $h(A,B,\pi)$ being used in step 3. Therefore, PAKA-X is secure against server compromise. •

4 Efficiency Analysis

Performance of key agreement protocols can be approximated in terms of communication and computation loads. PAKA is compared to the existing well-known protocols such as PAK [3], AKE [4], KS [5], and SNAPI [7] which are balanced password-authenticated key agreement protocols, while PAKA-X is compared to the existing

well-known protocols such as PAK-X[3], SNAPI-X[7], A-EKE[8], AMP[9], B-SPEKE[10], SRP[11] which are augmented password-authenticated key agreement protocols. Table 2 and 3 compare them regarding with several efficiency factors such as the number of protocol steps, random numbers, exponentiations, hash functions, and symmetric encryption/decryption, respectively.

Our protocol exchanges messages in 3 steps. However, the fact that the number of communication steps is small does not mean that the total execution time is also small. Protocols with message exchange of 3 steps depend on the protocol execution between two parties. That is, once each party receives the other's message, he/she is able to compute the response value using the message. For the measure of a total execution time, we will only consider modular exponentiations, which are the most time consuming operations in protocol. E(*Alice*:*Bob*) means parallel execution for modular exponentiation between both parties. That is, one party is able to compute something while he or she is waiting for the other party's reply. PAKA has 2E, that is, $E(g^{aQ} : g^{bQ^{-1}})$, $E((X_B \oplus Q^{-1})^{aQ} : (X_A \oplus Q)^{bQ^{-1}})$. Similarly, PAKA-X has 3E, that is, $E(g^a$: -), E(-: $(v)^b$), $E((X_B \oplus v)^{a \cdot h(A,B,\pi)^{-1}} : (X_A \oplus v)^b)$. Here '-' means no exponentiations.

As we can see in table 2 and 3, PAKA and PAKA-X are ones of protocols which have the smallest modular exponentiations for each party and the smallest total execution time compared to the well-known protocols although the number of communication steps is 4 steps more than 3 steps. Note that our protocols can be easily converted into the efficient protocols with 3 steps by only changing the flow of message exchange without modifying its nature for applications requiring small communication cost because of having the smallest modular exponentiations for each party.

Table 2 shows comparison of balanced password-authenticated key agreement protocols, in which two parties use a shared password.

Table 2. Comparison of balanced password-authenticated key agreement protocols

		SNAPI	AKE	KS	PAK	PAKA
# of steps		5	3	4	3	4
# of random numbers		3	2	4	2	2
# of exponentiations	*Alice*	3	2	4	3	2
	Bob	3	2	4	3	2
	Parallel	5	3	4	5	2
# of hash functions	*Alice*	3	3	4	4	3
	Bob	3	3	4	4	3
# of symmetric encryption /decryption		0	4	0	0	0

Table 3 shows comparison of augmented password-authenticated key agreement protocols.

Table 3. Comparison of augmented password-authenticated key exchange protocols

		SNAPI-X	A-EKE	B-SPEKE	SRP	AMP	PAK-X	PAKA-X
# of steps		5	5	4	4	4	3	4
# of random numbers		5	2	3	2	2	3	2
# of exponentiations	*Alice*	5	4	3	3	2	4	2
	Bob	4	4	4	3	3	4	2
	Parallel	7	6	6	4	3	8	3
# of hash functions	*Alice*	4	2	2	4	5	5	3
	Bob	3	1	2	3	4	5	3
# of symmetric encryption/decryption		0	4	4	0	0	0	0

5 Conclusion

The password scheme is the most widely used method because of its advantages of simplicity, convenience, adaptability, mobility, and less hardware requirement. The users just need to remember simple information like a password. This paper proposed a password-based authenticated key agreement protocol called PAKA, which provides mutual authentication and key agreement over an insecure channel only using a pre-shared password between two parties with low entropy. PAKA is secure against passive and active attacks, and provides perfect forward secrecy. We then extend PAKA to a protocol called PAKA-X, in which the client stores a plaintext version of the password, while the server stores a verifier for the password. PAKA-X has an additional security in the case of server compromise. The security of them is based on the difficulty of Diffie-Hellman problem and discrete logarithm problem. Our protocols are very simple in structure and efficient in the perspective with total processing time.

Acknowledgements. This work was supported by the Brain Korea 21 Project in 2003.

References

1. S. Bellovin and M. Merritt. Encrypted key exchange: Password-based protocols secure against dictionary attacks. In Proceedings of IEEE security and Privacy, (1992) 72-84
2. IEEE. Standard Specifications for Public Key Cryptography, IEEE1363, (2002)
3. V. Boyko, P. MacKenzie and S. Patel. Provably Secure Password-Authenticated Key Exchange Using Diffie-Hellman, Advances in Cryptology-EUROCRYPT'2000, (2000) 156-171
4. M. Bellare, D. Pointcheval and P. Rogaway. Authenticated Key Exchange Secure Against Dictionary Attacks, Advances in Cryptology-EUROCRYPT'2000, (2000) 139-155

5. T. Kwon and J. Song. A Study on the Generalized Key Agreement and Password Authentication Protocol, IEICE TRANS. COMMUN., Vol. E83-B, No. 9, (2000) 2044-2050
6. P. MacKenzie and R. Swaminathan. Secure network authentication with password identification, Presented to IEEE P1363a, (1999)
7. P. MacKenzie, S. Patel, and R. Swaminathan. Password-authenticated key exchange based on RSA. In ASIACRYPT2000, (2000)
8. S. Bellovin and M. Merritt. Augmented encrypted key exchange: a password-based protocol secure against dictionary attacks and password-file compromise, ACM Conference on Computer and Communications Security, (1993) 244-250
9. T. Kwon. Ultimate Solution to Authentication via Memorable Password, Presented to IEEE P1363a, (2000)
10. D. Jablon. Extended password key exchange protocols, WETICE Workshop on Enterprise Security, (1997)
11. T. Wu. Secure remote password protocol, Internet Society Symposium on Network and Distributed System Security, (1998)
12. T. Kwon and J. Song. Secure agreement scheme for gxy via password authentication, Electronics Letters, Vol. 35, No. 11, (1999) 892-893
13. M. Bellare and P. Rogaway. Entity Authentication and Key Distribution, Advances in Cryptology-CRYPTO'93, Vol. 773, (1994) 232-249

A Two-Public Key Scheme Omitting Collision Problem in Digital Signature*

Sung Keun Song[1], Hee Yong Youn[1], and Chang Won Park[2]

[1]School of Information and Communications Engineering
Sungkyunkwan University, 440-746, Suwon, Korea
kkskk103@skku.edu, youn@ece.skku.ac.kr

[2]IT System Research Center
Korea Electronics Technology Institute, Korea
parkcw@keti.re.kr

Abstract. Since the concept of digital signature was ever introduced, a number of digital signature algorithms have been proposed. Although various digital signature algorithms are available, connoted hazardous factors in terms of security are still existing. Among them collision problem associated with the hash algorithms used for signing is a significant obstacle needed to be overcome. In this paper we develop a method solving the problem and propose a new and secure digital signature scheme. It is achieved by using an additional public key on top of the existing PKI system. The new digital signature scheme can be used without reconstructing the structure of the existing digital signature mechanism. Therefore, we can flexibly select the new scheme or existing one according to the required degree of security. It is also effective for mutual signature.

1 Introduction

Since the concept of digital signature was ever introduced, a number of digital signature algorithms have been proposed. The representative algorithms are RSA [1], ElGamal [2], and DSS (Digital Signature Standard) algorithm [3]. Although various digital signature algorithms are available, connoted hazardous factors in terms of security are still existing. Among them collision problem associated with the hash algorithms used for signing is a significant obstacle needed to be overcome.

The security of digital signature directly depends on the security of public key cryptography because it is based on a theory of cryptography. Similarly, digital signature algorithm displays various level of security according to how to use the hash algorithm. The security of the hash algorithm is seriously affected by collision problem. Various hash algorithms have been developed along with the digital signature

* This work was supported in part by 21C Frontier Ubiquitous Computing and Networking, Korea Research Foundation Grant (KRF - 2003 - 041 - D20421) and the Brain Korea 21 Project in 2003. Corresponding author: Hee Yong Youn

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 627–636, 2004.
© Springer-Verlag Berlin Heidelberg 2004

schemes proposed in the literature. Even though it is not easy for an adversary to attack a message tagged with a digital signature by taking advantage of the hash collision problem, it is still possible that an adversary counterfeits the message. Here, an important problem is that if a message is counterfeited, one cannot prove illegality of the counterfeited message. This hash collision problem may cause a devastating result, especially when the message includes critical information.

Therefore, in this paper, we develop a method solving the problem and propose a new and secure digital signature scheme. It is achieved by using an additional public key on top of the existing PKI system. The new digital signature scheme can use a hash algorithm which allows fast operation while providing high security. The new digital signature scheme can be used without reconstructing the structure of the existing digital signature scheme. Therefore, we can flexibly select the new scheme or the existing one according to the required degree of security. It is also effective for mutual signature. We provide detail operational mechanism of the proposed signature scheme and analyze its security.

The rest of the paper is organized as follows. Section 2 presents a brief description of digital signature. Section 3 investigates the vulnerability of digital signature due to collision problem of the hash algorithm. Section 4 proposes a new digital signature scheme, and the security of the scheme is evaluated. Finally, we conclude the paper in Section 5.

2 Digital Signature

A digital signature is a pair of large numbers represented as strings of binary digits. It is computed using a set of rules and parameters with which identity of the signatory and integrity of the data can be verified. Here, an algorithm is used to provide the way how to generate and verify the signature. The signature generation process makes use of a private key to generate a digital signature, while the signature verification process makes use of a public key corresponding to the private key. Each user possesses a private and public key pair which are different. Public keys are assumed to be known to the public by certification of certificate authority (CA) in general. Private keys are never shared. One can verify the signature of a user by using the user's public key. Only the possessor of a private key can generate signatures as long as the key has not been revealed.

A hash algorithm is used in the signature generation process to obtain a condensed version of message, called a message digest. The message digest is then input to the digital signature algorithm to generate a digital signature. The digital signature is sent to the intended verifier along with the message. The verifier of the message and signature verifies the signature by using the sender's public key.

The same hash algorithm as the one used by the sender must be used in the verification process. The hash algorithm is specified in a separate standard, the Secure Hash Standard, FIPS 180-1 [4]. FIPS approved digital signature algorithms implemented with the Secure Hash Standard. Similar procedures may be used to generate and verify signatures for stored as well as transmitted data.

3 Vulnerabilities of the Hash Algorithm

As explained above, digital signature algorithms have two connoted hazardous factors in terms of security. They are the inherent limitation of digital signature algorithm and the collision problem of hash algorithm used for signing. The security of a digital signature algorithm depends on the security of public key cryptography. The collision problem of the hash algorithm, the second hazardous factor, is another factor limiting the security of digital signature.

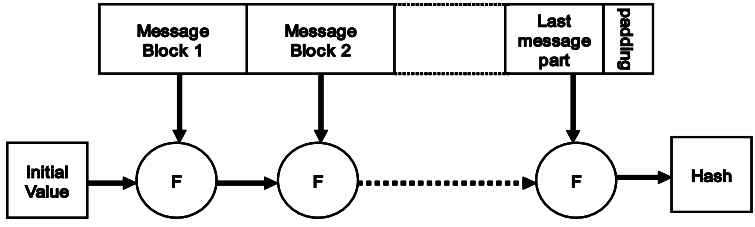


Fig. 1. Merkle-Damgard model.

A hash algorithm maps an arbitrary-length message to a fixed-length value, which must be fast to be practical. On the other hand, the hash algorithm must be collision-resistant, i.e. it must be computationally infeasible to find a collision, which is a pair of different messages with the same hash value. However, collision cannot be avoided. MD5, SHA, and RIPEMD-160 are representative hash algorithms [5-8]. Many of the existing hash algorithms follow a design principle of Merkle-Damgard [9] shown on Fig. 1. Essentially, this model simplifies the management of large inputs and production of a fixed-length output by using a function *F*, which is usually called a compression function. Given a compression function, a hash algorithm can be defined as repeated applications of the function until the entire message has been processed.

In this process a message of arbitrary length is broken into blocks whose length depends on the compression function, and padded so that the size of the message becomes a multiple of the block size. The blocks are then processed sequentially, taking the result of hashing so far and the current message block as inputs, with the final output being the hash value of the entire message. The hash function is repeatedly applied to each message block and hash value of the previous blocks. The security of this scheme rests on the security of the *F* function. Note that the more the message size increases, the more the number of collisions per hash value increases exponentially. For example, assume that one message block is 512 bits and the *F* function returns 128-bit output. When a message has one block, the number of collision is

$$\frac{1\times 2^{512}}{2^{128}} = 2^{384}$$

When a message has two blocks, the number of collision is

$$\frac{2^{128}\times 2^{384}\times 2^{512}}{2^{128}} = 2^{896}$$

For a message of three blocks,

$$\frac{2^{128}\times 2^{896}\times 2^{512}}{2^{128}} = 2^{1408}$$

In general, when a message has n blocks, it is

$$\frac{2^{128}\times 2^{512n-640}\times 2^{512}}{2^{128}} = 2^{512n-128}$$

Because of this property, a malicious signatory, verifier, or third party can counterfeit a message which is signed by a signatory by adding new strings, words, or blanks etc. to the original message to exploit the collision problem of the hash algorithm. Here, the digital signature of a counterfeited message as well as the TS (Time Stamp) token are valid. The signatory requests TSS (Time Stamping Service) by sending the hash value of the message to the TSA (Time Stamping Authority) [10]. Therefore, a signatory cannot prove that the counterfeited message was not signed by itself. It is due to the collision problem of the hash algorithm, and many crucial problems may occur in e-commerce and practical applications if that happens.

We can classify the attacks taking advantage of the collision problem of a hash algorithm into three types.

- An attacker researches the structural weakness of the hash algorithm to identify collision.
- An attacker accumulates digital signature and TS token corresponding to each hash value for the life time of a certificate. For counterfeits, the attacker finds a digital signature from the database whose hash value is equal to that of the counterfeited message.
- An attacker counterfeits the message by modifying the counterfeited message until the hash value of the counterfeited message is same as that of the target message.

The third attack above can be classified by two viewpoints. First is the counterfeit method in the viewpoint of the *verifier*. Malicious verifier would try to counterfeit a received message using the collision problem by modifying the received message until the hash value of the counterfeited message is same as that of the original message as follows.

$$h(M)=h(M') \ (M' : \text{the counterfeited message})$$

Thereafter, the digital signature and TS token of the original message are attached to the counterfeited message. Here, the digital signature of the counterfeited message is valid because the hash values of the original message and the counterfeited message are same. Also, TS of the counterfeited message is valid. If the message is an important document of an e-commerce, for example, this may be a crucial problem to the signatory.

Second is the counterfeit method in the viewpoint of the *signatory*. There may be cases that two users need to sign a message as a contract in an e-commerce. In this case, if the signatory (User-A) tries to counterfeit the message maliciously that is signed by a person and the verifier (User-B), User-A can counterfeit the message as follows. First, User-A modifies the counterfeited message so that the hash value of it

is same as that of the original message, and then sends the original message, a digital signature, and TS token to User-B. User-B may sign normally on the received message, exclude the digital signature of User-A after it is confirmed, and then return the message to User-A. When User-A receives the original message that is signed by User-B, it separates the digital signature and TS token from the message. User-A combines the counterfeited message with the digital signature of User-A and User-B, and now the original message is counterfeited. The digital signatures and TS token of the message are still valid. As a result, a great damage may occur to User-B if the counterfeited message includes some important document. Fig. 2 shows the counterfeit method in the viewpoint of the signatory. We next present a new scheme which can avoid this kind of problem and thereby allowing a much more secure digital signature.

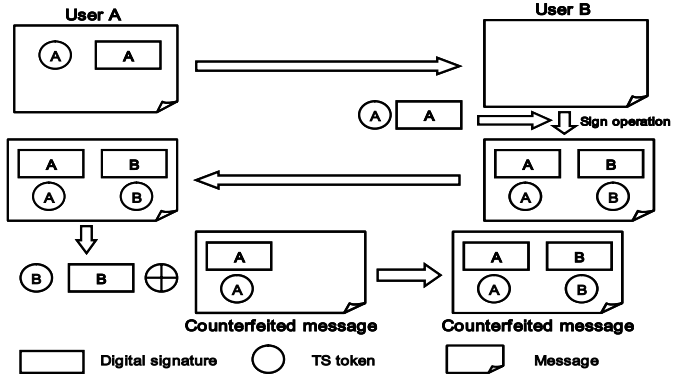


Fig. 2. The counterfeit method in the viewpoint of the *signatory*.

4 The Proposed Digital Signature Scheme

So far, we have explained the forgery problem of existing digital signature algorithms due to the collision problem. This section proposes a new digital signature scheme solving the problem. It uses a cryptographic algorithm, which employs two different public keys. In this paper we call it "two-public key cryptography". The basic idea is to hide the hash value of a message from the verifier using the two-public key cryptography. The validity of the digital signature of a message is confirmed by a trustworthy entity such as Certificate Authority in the public key infrastructure. First, we explain the two-public key cryptography. Then, we propose the new digital signature scheme.

4.1 The Two-Public Key Cryptography

Fig. 3 shows the structure of the proposed two-public key cryptography. Note that if a private key is used to encrypt something using Algorithm-B, only public key-2 can decrypt it. That is, according to the algorithm used for encryption, the public key that can decrypt the message varies.

We show an example of two-public key cryptography using the RSA and ElGamal scheme, the two representative public key cryptography algorithms. First, we review the two.

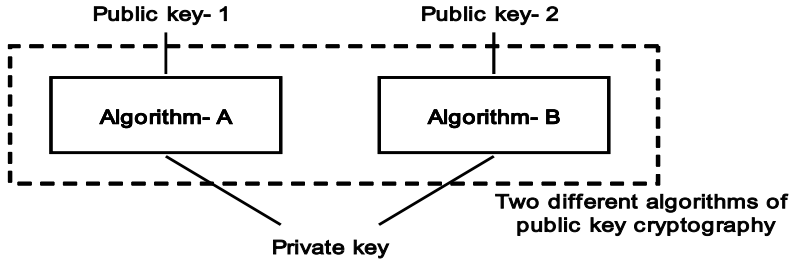


Fig. 3. The structure of two-public key cryptography.

The RSA cryptography, named after its inventors R. Rivest, A. Shamir, and L. Adleman, is the most widely used public key cryptography. It may be used to provide both secrecy and digital signatures, and its security is based on the intractability of the integer factorization problem. Each user creates a RSA public key and the corresponding private key. The usersψdo the following [1]:

1. Generate two large random (and distinct) primes $p\psi$and q, each roughly the same size.
2. Compute $n=pq\psi$and $\phi=(p-1)(q-1)$
3. Select a random integer e, $1< e < \phi$, such that $\gcd(e, \phi)=1$.
4. Use the extended Euclidean algorithm to compute the unique integer d, $1< d < \phi$, such that $ed \equiv 1 \pmod{\phi}$.
5. The public key is (n, e); private key is d.

The ElGamal public-key encryption scheme can be viewed as Diffie-Hellman key agreement in the key transfer mode. Its security is based on the intractability of the discrete logarithm problem and the Diffie-Hellman problem. Each user creates a public key and the corresponding private key. The usersψdo the following [2]:

1. Generate a large random prime $p\psi$and a generator $\alpha\psi$of the multiplicative group $Z_p\psi$of the integers modulo p.
2. Select a random integer a, 1 a p-2, and compute $y= \alpha^a \bmod p$.
3. The public key is (p, α, y); private key is a.

In above we can recognize that if the prime p of the ElGamal and the ϕ of the RSA scheme have a same value, the private keys of them become same. If the a of the ElGamal is denoted by the d of the RSA, public key-1 is (n, e), public key-2 is (p, α, y), and the common private key is d in the proposed two-public key cryptography. In this way we can construct two-public key cryptography using the RSA and ElGamal scheme. Of course, we can easily construct various two-public key cryptography schemes using any two different public key cryptographies.

4.2 The New Digital Signature Scheme

The newly proposed digital signature algorithm is based on PKI as in the existing digital signature schemes. In general, a CA that is one of the components of the PKI is responsible for creating and issuing end-entity certificates, and management of all aspects of the life cycle of a certificate after its issuance [11]. In the new digital signature scheme, we add functions that confirm the validity of a digital signature of a signatory and then sign the message using a private key for certifying the CA. A verifier confirms the validity of a digital signature of a signatory by confirming the digital signature of the CA. The new digital signature scheme consists of three processes; signature generation process, certificate process, and verification process.

P_A: a public key of algorithm-A; be known to all objects of PKI
P_B: a semipublic key of algorithm-B; be known to only the CA's
P_{AB}^{-1}: a private key of the two-public key cryptography
$\{\ \}AP_{AB}^{-1}$: encrypt or decrypt the private key using algorithm-A
K: a random key of a symmetric cryptography
H: a hash function that extends the input regardless of its value
h: a hash function that reduces the input regardless of the message extent

A signature generation process handled by a signatory (User-A) is as follows. First, the signatory generates a random key, K, and then encrypts a message using it so that a CA cannot see the message. If the content of the message is not important, it is unnecessary to encrypt the message. The signatory calculates a hash value of the random number (RN), H(RN). Here, the extent of the H(RN) has one block size of symmetric cryptography. The signatory calculates a hash value where the H value is added to the encrypted message, h({M}K, H(RN)). When h is calculated, the H value is put on a specific block of the encrypted message that User-A selected. Even though the message is not encrypted, the H value is still put on the block decided by the signatory. The signatory generates a digital signature by encrypting the h value, block position, and the random number using algorithm-B and its own private key. The signatory requests TS token by sending a hash value, h({M}K) to TSA. Thereafter, the signatory sends the encrypted message, the digital signature, an ID, etc. to the CA as follows. Fig. 4 shows the signature generation process.

$$\text{ID A, } \{M\}K, \text{ User-A}_{\text{Digital Signature}}, \{K\}AP_{\text{A-User B}}, \text{ TS token} \rightarrow \text{CA}$$

$$\text{- User-A}_{\text{Digital Signature}}\text{: } \{h(\{M\}K, H(RN)) \| \text{block position} \| RN\}\ BP^{-1}{}_{\text{User A}}$$

The certificate process handled by a CA is as follows. The CA searches a semipublic key of User-A, P_B, from a database using the ID of User-A, and then decrypts the digital signature to obtain the block position and RN. Thereafter, the CA calculates a hash value, h({M}K, H(RN)), by using the block position and RN, and then compares the hash value with the h({M}K, H(RN)) value which is part of the decrypted digital signature, and verifies TS token. If the values are same and the TS token is valid, the CA calculates a hash value, h(SN‖{M}K) where the sign number(SN) is added to the front of {M}K, and then calculates User-$A_{\text{Digital Signature}}$⊕h(SN‖{M}K).

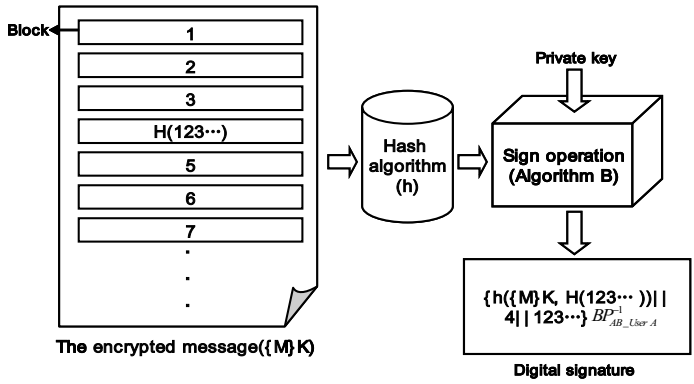


Fig. 4. The signature generation process.

The CA generates a digital signature by encrypting User-$A_{\text{Digital Signature}}$⊕h(SN‖{M}K) by its own private key for certifying the digital signature of User-A. Although the asymmetric cryptography used by the CA is not based on the two-public key cryptography, it does not become a problem. The CA sends messages received from User-A, its own digital signature for certificate of User-A, SN and certificate to the verifier as follows.

$$\text{SN, } \{M\}K, \{K\}\ AP_{\text{A-User B}}, \text{ TS token, User-}A_{\text{Digital Signature}}, \text{ CA}_{\text{Digital Signature}},$$
$$\text{CA's Certificate} \rightarrow \text{User-B}$$
$$\text{- CA}_{\text{Digital Signature}}\text{: } \{h(SN\|\{M\}K) \oplus \text{User-}A_{\text{Digital Signature}}\}\ P^{-1}_{\text{CA}}$$

Here, the CA cannot see the message because it does not know the random key, K, used to encrypt the message.

The verification process handled by a verifier (User-B) is as follows. First, the verifier verifies validity of TS token and calculates h(SN‖{M}K). Thereafter, the verifier decrypts the digital signature of the CA using the public key of the CA, and calculates the decrypted digital signature ⊕ User-$A_{\text{Digital Signature}}$. The verifier compares h(SN‖{M}K) of the decrypted digital signature with the calculated h(SN‖{M}K) by itself. If the values are same, the digital signature of the signatory is valid. Therefore, the encrypted message is also valid. The verifier decrypts the encrypted K using the verifier's own private key, and then obtains the message by decrypting the encrypted message using K.

The case of mutual signature where two users need to sign the message as a contract is as follows. Mutual signature requires the same process as above until the verification process begins. In the remainder of the mutual signature, the verifier requests TS token by sending a hash value, h({M}K) to TSA, signs the received message like the signature process of the signatory, and then sends the digital signature and message, ID, etc. to the CA as follows.

$$\text{ID B, User-}A_{\text{Digital Signature}}, \text{ CA}_{\text{Digital Signature}}, \text{ User-}B_{\text{Digital Signature}}, \{M\}K,$$
$$\text{TS token A, TS token B} \rightarrow \text{CA}$$
$$\text{- User-}B_{\text{Digital Signature}}\text{: } \{h(\{M\}K, H(RN')\|\text{block position}'\|RN'\}\ BP^{-1}{}_{\text{User A}}$$

The CA verifies validity of the digital signature and TS token of the verifier like the certificate process above. If the digital signature of the verifier is valid, the CA

calculates the hash value where the digital signature of the signatory and verifier are added to the sign number, $h(SN||User\text{-}A_{Digital\ Signature}||User\text{-}B_{Digital\ Signature})$, and then signs the message by encrypting the hash value using its own private key for certificate of the mutual signature after removing the existing digital signature of the CA. Herewith, the message may come into effect. Thereafter, the CA sends the message, the digital signatures, the certificate of the CA's own, etc. to the signatory and the verifier as follows.

$SN, User\text{-}A_{Digital\ Signature}, CA_{Digital\ Signature}, User\text{-}B_{Digital\ Signature}$, TS token A, TS token B, $\{M\}K$,
CA's Certificate $\rightarrow$User-A, User-B
- $CA_{Digital\ Signature}$: $\{h(SN||User\text{-}A_{Digital\ Signature}||User\text{-}B_{Digital\ Signature})\}\ P^{-1}{}_{CA}$

4.3 Security of the Proposed Scheme

The new proposed digital signature scheme solves the problems presented in Section 3. That is, any object of PKI cannot counterfeit a message by taking advantage of the collision problem of the hash algorithm in the proposed digital signature scheme. Without the proposed scheme a verifier can counterfeit a message using the collision problem. With our scheme, the verifier cannot do that since the verifier is not able to know the RN and block position of the signature.

A signatory may counterfeit a message in mutual signature. This is not possible with the proposed scheme. Even though the signatory knows the RN, block position, and the hash value of the signatory, the signatory cannot counterfeit the message because the signatory does not know the hash values of the verifier. That is, even though the signatory can counterfeit the message such that the digital signature of the signatory is valid, the signatory cannot make that the digital signature of the verifier valid. Note that the RN and block position of the signatory and those of the verifier are not same. Accordingly, the h value of the verifier is not same as the new h value of the counterfeited message.

$$h(\{M\}K, H(RN_{verifier})) \quad h'(\{M'\}K, H(RN_{verifier}))$$

In this way the signatory or verifier cannot counterfeit the message. The CA cannot counterfeit the message either because it does not know K and the message contents encrypted using the K. Therefore, the security of the new digital signature scheme is not limited by the security of hash algorithms.

Is the two-public key cryptography more secure than public key cryptography? It is difficult to answer the question. This is because security of any cryptographic algorithm is influenced by many factors such as difficulty of the mathematical problem of the cryptographic algorithm, complexity of the cryptographic algorithm, and key length, etc. However, we can expect the following. If the securities of the two cryptographic algorithms employed in the two-public key cryptography are similar, the security of a two-public key cryptography is similar to the security of each of the two cryptographic algorithms since they are based on different problem of mathematics. Therefore, a system designer must design the two-public key cryptography using two different public key cryptographies of the same level of security.

5 Conclusion

In this paper we have proposed a new digital signature scheme solving the collision problem of hashing required in the existing digital signature algorithms. As a result, security of the new digital signature scheme is not limited by the security of the hash algorithm. The new digital signature scheme can use a hash algorithm which allows fast operation while providing high security. It can also be used without reconstructing the structure of the existing digital signature scheme. Therefore, we can flexibly select the new scheme or the existing one according to the degree of required security. When a securer signature is required, the new scheme is selected. Otherwise, the existing algorithm is selected. We anticipate that the new digital signature scheme can significantly promote e-commerce by increasing the security of transactions. In the future we plan to investigate the performance of the proposed scheme using various combinations of public key cryptographies.

References

1. Rivest, R., Shamir, A., Adleman, L.: A Method for Obtaining Digital Signatures and Public Key Cryptosystems. Communications of the ACM, (1978) 120-126
2. ElGamal, T.: A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Info. Theory, IT-31, No. 4, (1985) 469-472
3. National Institute of Standards and Technology (NIST).: Digital Signature Standard. FIPS PUB 186-2, (2000) http://csrc.nist.gov/publications/ fips/fips186-2/fips186-2-change1.pdf
4. National Institute of Standards and Technology (NIST).: Secure Hash Standard. FIPS PUB 180-1, (1995) http://www.itl.nist.gov/fipspubs/fip180-1.htm
5. Dobbertin, H.: The status of MD5 after a recent attack. RSA Laboratories, CryptoBytes, 2(2), (1996)
6. Eastlake, D., 3rd., Jones, P.: US Secure Hash Algorithm 1 (SHA1). RFC 3174, (2001) http://www.faqs.org/rfcs/rfc3174.html
7. Keromytis, A., Provos, N.: The Use of HMAC-RIPEMD-160-96 within ESP and AH. RFC 2857, (2000)
8. Dobbertin, H., Bosselaers, A., Preneel, B.: RIPEMD-160: a strengthened version of RIPEMD. Fast Software Encryption, LNCS 1039, D. Gollmann, Ed., Springer-Verlag, (1996) 71-82
9. Damgard, I.B.: A design principle for hash functions. Advances in Cryptology-Crypto '89, Lecture Notes in Computer Science, vol. 435, Springer-Verlag, (1990) 416-427
10. Adams, C., Cain, P., Pinkas, D., Zuccherato, R.: Internet X.509 Public Key Infrastructure Time Stamp Protocol, (1998) draft-ietf-pkix-time-stamp-00.txt
11. Housely, R., Ford, W., Polk, W., Solo, D.: Internet X.509 Public Key Infrastructure. IETF RFC 2459. (1999)

A Novel Data Encryption and Distribution Approach for High Security and Availability Using LU Decomposition*

Sung Jin Choi and Hee Yong Youn

School of Information and Communications Engineering
Sungkyunkwan University, Suwon, Korea
{choisj, youn}@ece.skku.ac.kr

Abstract. As the society increasingly relies on digitally stored and accessed information, supporting the availability, persistence, integrity, and confidentiality of the information becomes more critical. Mirroring is a simple but practical approach allowing reasonable availability and accessibility, but it cannot provide security. This paper proposes an approach which integrates data encryption and distribution to allow high security while providing the same degree of availability as mirroring. It is based on LU decomposition of matrix. Analysis confirms that it allows slightly higher availability compared with mirroring scheme for both read and write operation. The proposed scheme needs almost the same storage space as mirroring.

1 Introduction

As the modern society increasingly relies on digitally stored and accessed information, supporting the availability, persistence, integrity, and confidentiality of the information becomes more critical. Furthermore, with the continuing shift towards pervasive computing and less-expert users/administrators, the information storage infrastructures must be more self sufficient. Mirrored data storage system provides a solution for this despite failures and malicious compromises of storage nodes, client systems, and user accounts [1].

Mirrored data storage system can survive failures and compromises of storage nodes by storing data at a set of nodes via well-chosen encoding and mirroring schemes. Many such schemes have been proposed and employed over the years, but little understanding exists on the trade-off that they comprise. The selection and parameterization of the data distribution scheme has a profound impact on the availability, security, and performance of the mirrored data storage system [2-3].

Data distribution is one of the key technologies developed for achieving a desired level of security and availability, and it involves data encoding and partitioning algorithm. There exist many such algorithms applicable to data distribution, including

* This work was supported in part by 21C Frontier Ubiquitous Computing and Networking, Korea Research Foundation Grant (KRF - 2003 - 041 - D20421) and Brain Korea 21 Project in 2004. Corresponding author : Hee Yong Youn

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 637–646, 2004.
© Springer-Verlag Berlin Heidelberg 2004

encryption, mirroring, striping, erasure-resilient coding, secret sharing, and various combinations of them. They offer different levels of performance (throughput), availability (probability that data can be accessed), and security (effort required to compromise the confidentiality or integrity of stored data). For example, mirroring provides high availability at the sacrifice of network bandwidth and storage space, whereas short secret sharing provides high availability and security with low storage and network bandwidth overhead but with high computation overhead. Likewise, selecting the number of shares required to reconstruct a secret-shared value involves a trade-off between availability and confidentiality; if more machines must be compromised to steal the secret, then more machines must be operational to provide it legitimately. Availability of mirrored data storage systems needs to be improved through data distribution, and thus development of new efficient data distribution and protection scheme is inevitable [4].

In mirrored data storage systems the availability and security of data are greatly influenced by the policy how the data are dispersed. For example, complete recovery is possible even though most data are lost in one scheme while failure of one node causes loss of entire data in another scheme, vice versa. Usually, the requirements for achieving high availability and security with storage system conflict with each other, and there exist some tradeoff between them. This paper proposes a novel data dispersal/encryption scheme to allow highly secure storage system by using some important properties of LU (Low, Upper triangular) decomposition of matrix, while providing a same degree of availability as mirroring. Simple mirroring cannot provide security. Analysis confirms that it allows slightly higher availability compared with mirroring scheme for both read and write operation. The proposed scheme needs almost the same storage space as mirroring.

The rest of the paper is organized as follows. The next section explains representative data distribution schemes. Section 3 presents the proposed dispersal/encryption scheme based on LU decomposition of matrix. In Section 4 we evaluate the availability of the proposed scheme and compare it with mirroring. Finally, Section 5 concludes the paper with future work.

2 Related Work

Mirrored data storage systems requiring high data availability and security usually employ a scheme to encode data before they are stored. Specifically, a *p-m-n* threshold scheme breaks data into *n* shares such that any *m* of the shares can reconstruct the original data and fewer than *p* shares reveal absolutely no information on the original data. Although encryption makes it difficult to extract the original data, it does not change the value of *p* since information is still available for theft. Different parameter selections of *p-m-n* expose a large space of encoding mechanisms for storage. The threshold schemes can be used instead of cryptographic techniques to guarantee the confidentiality, and several schemes can be combined [5-6].

An example of a specific threshold scheme is *N*-way mirroring, which is a *1-1-N* threshold scheme. That is, each replica reveals some information on the encoded data

(hence $p = 1$). A single replica is required to reconstruct the original data ($m = 1$), and there are N replicas to select from when attempting to reconstruct the original data ($n = N$).

Disk mirroring based on N-way mirroring is to provide an up-to-date mirror image of critical business data available on duplicate disks at all times. The disks may reside in the same system or in different systems. The secondary volumes may reside locally with or remotely from the primary volume. The duplicate disks may be used to facilitate data migrations, replicate production data for development and testing, or simply provide a substitute in case disk failures occur in the primary system. By storing the mirrored image remotely, the network may also provide a secure disaster recovery solution. Access to duplicate data is available almost immediately, thereby minimizing the duration and impact of downtime [7-8].

There are also important data distribution algorithms outside the class of p-m-n threshold schemes. Notably, encryption is a common approach to protecting the confidentiality of information. Symmetric key encryption (e.g., triple-DES, AES) is a data distribution algorithm characterized by a single parameter key length. Hybrid data distribution algorithms can be constructed by combining the algorithms already discussed. For example, many mirrored data storage systems combine replication with encryption to address availability and security, respectively. Security in such a system hinges both upon how well the encryption keys are protected and upon the difficulty of cryptanalysis on the information gained by collecting the shares because the information gained pertains to the encrypted data [9-10].

As another example, short secret sharing encrypts the original value with a random key, stores the encryption key using secret sharing, and stores the encrypted information using secret sharing or information dispersal. Short secret sharing algorithms have three parameters: m, n, and k (key length). Public key cryptography (e.g., RSA) can be used instead of symmetric key cryptography to protect information confidentiality. Short secret sharing offers a different set of trade-offs between confidentiality and storage requirements from general threshold schemes. Management of cryptographic keys must be addressed in the design of a system that uses cryptography; symmetric key and public key cryptography require different key management strategies. Finally, compression algorithms (e.g., Huffman coding) can be used before the data are distributed to reduce the size of the data that must be encoded.

Another important type of data distribution algorithm provides integrity verification. Cryptographic hash algorithms (e.g., MD5, SHA-1) can be used to add digests to the data before they are encoded with another algorithm. The hash value of the decoded data can be compared with the digest to verify the integrity of the decoded data. Digests can also be generated for shares resulting from an encoding (e.g., distributed fingerprints [11]), allowing integrity to be verified prior to decoding the data. Hash algorithms are parameterized by the hash size, and they must either be encoded with or stored separately from the data in order to be effective. Digital signatures (e.g., DSA) can provide similar integrity guarantees as hash algorithms.

Two classes of integrity algorithms are often used to build authentication and directory services that protect the integrity of meta-data. The first class includes agreement algorithms such as the Byzantine fault tolerant algorithm [12]. Quorum systems which are a superset of voting algorithms belong to the second class. Also, there are

integrity checking algorithms that work exclusively with threshold algorithms. In the schemes where *m* is smaller than *n*, excess shares can be used during decoding (different permutations of *m* shares are used for validation). Secret sharing schemes can also be modified directly to offer a probabilistic guarantee of cheater detection [13-14].

Among various threshold schemes mirroring is quite simple and practical, and therefore it is widely used in practice. We next present the proposed scheme which improves mirroring in terms of both security and availability without increasing the storage space.

3 The Proposed Scheme

3.1 Dispersal and Encryption Schemes

We first lay out the data in matrix format before applying the proposed dispersal/encryption scheme using LU decomposition. A matrix is represented as follows.

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

To decompose a matrix for dispersal and encryption, we first develop some definitions and theorems.

Definition 1: (Elementary Matrix) An $n \times n$ matrix is called elementary if it can be obtained from the identity matrix I_n by using one and only one elementary row operation (elimination, scaling, or interchange). Hence, elementary matrix is always equivalent to I_n. Elementary row operations are $R_i \leftrightarrow R_j$, $cR_i \leftrightarrow R_i$, $R_i + cR_j \leftrightarrow R_i$ [15-16].

Theorem 1: A~B if and only if there are elementary matrix $E_1,...,E_k$ such that $B = E_k \cdots E_1 A$. In particular, Theorem 1 applies to a matrix A and an echelon form U of A.

Proof 1: If U is obtained from A by using row operations with corresponding elementary matrix $E_1,...,E_k$, then $U = E_k \cdots E_1 A$. Solving for A yields $A = (E_k \cdots E_1)^{-1} U$ or $A = E_1^{-1} \cdots E_k^{-1} U$ [13-14] ■.

We begin with an $n \times n$ matrix A and search for a matrix using Definition 1 and Theorem 1

$$L=\begin{bmatrix} l_{11} & 0 & 0 & \cdots & 0 \\ l_{21} & l_{22} & 0 & \cdots & 0 \\ l_{31} & l_{32} & l_{33} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & l_{n3} & \cdots & l_{nn} \end{bmatrix},\ U=\begin{bmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ 0 & u_{22} & u_{23} & \cdots & u_{2n} \\ 0 & 0 & u_{33} & \cdots & u_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & u_{nn} \end{bmatrix}$$

such that

$$A=LU$$

$$\text{where } L=E_1^{-1}E_2^{-1}\cdots E_n^{-1},\ U=E_nE_{n-1}\cdots E_1A \tag{1}$$

When this is possible, we say that A has an LU-decomposition. Note that L and U are not uniquely determined by Equation (1). In fact, for each i, we can assign a nonzero value to either l_{ii} or u_{ii} (but not both). For example, one simple choice is to set $l_{ii}=1$ for i=1, 2,…, n, thus making L unit lower triangular. Another obvious choice is to make U unit upper triangular ($u_{ii}=1$ for each i). The L and U matrix obtained are the data actually stored in two separate nodes respectively.

Example 1: If original data is $A=\begin{bmatrix} 2 & 1 & 1 \\ 4 & 1 & 0 \\ -2 & 2 & 1 \end{bmatrix}$, then we calculate $E_1=\begin{bmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, $E_2=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$, $E_3=\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 3 & 1 \end{bmatrix}$ using Definition 1 and Theorem 1. Therefore, the data actually stored in each node are as follows.

$$L=E_1^{-1}E_2^{-1}E_3^{-1}=\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ -1 & -3 & 1 \end{bmatrix},\ U=E_3E_2E_1A=\begin{bmatrix} 2 & 1 & 1 \\ 0 & -1 & -2 \\ 0 & 0 & -4 \end{bmatrix}$$

An important property of the proposed scheme is that it allows both secret dispersal as a general threshold scheme and encryption of data at the same time. That is, one cannot extract original data even though the stored data, L or U matrix, are available since deciding the original matrix using L or U matrix is NP-hard problem. Therefore, the proposed scheme provides high security which simple mirroring cannot at all. The availability is also slightly higher than mirroring scheme as shown later. Note that the proposed scheme needs almost the same storage space as mirroring.

3.2 Data Recovery

To derive a data recovery algorithm using the L U matrix of A, we start with the formula for matrix multiplication.

$$a_{ij} = \sum_{s=1}^{n} l_{is} u_{sj} = \sum_{s=1}^{\min(i,j)} l_{is} u_{sj} \,. \tag{2}$$

Here we have used the fact that $l_{is} = 0$ for $s > i$ and $u_{sj} = 0$ for $s > j$.

Each step in this process determines one new row of U matrix and one new column of L matrix. At Step k, we can assume that the rows 1, 2,…, k-1 of U matrix and columns 1, 2,…, k-1 of L matrix have already been obtained. Putting i = j = k in Equation (2), we obtain

$$a_{kk} = \sum_{s=1}^{k-1} l_{ks} u_{sk} + l_{kk} u_{kk} \,. \tag{3}$$

If u_{kk} or l_{kk} has been specified, we use Equation (3) to determine the other elements. With u_{kk} or l_{kk} known, we use Equation (2) to derive the kth row (i=k) and the kth column (j=k), respectively,

$$a_{kj} = \sum_{s=1}^{k-1} l_{ks} u_{sj} + l_{kk} u_{kj} \quad (k+1 \le j \le n). \tag{4}$$

$$a_{ik} = \sum_{s=1}^{k-1} l_{is} u_{sk} + l_{ik} u_{kk} \quad (k+1 \le i \le n) \tag{5}$$

If $l_{kk} \neq 0$, Equation (4) can be used to obtain elements u_{kj}. Similarly, if $u_{kk} \neq 0$, Equation (5) can be used to obtain elements l_{ik}. It is interesting to note that these two computations can be carried out in parallel. The pseudo-code for the proposed scheme is as follows:

```
input n
for k=1 to n do
Specify a nonzero value for either
a_kk, a_kj, a_ik compute the elements from
a_kk = l_kk u_kk + Σ_{s=1}^{k-1} l_ks u_sk
for j=k+1 to n do
        a_kj ← l_kk u_kj + Σ_{s=1}^{k-1} l_ks u_sj
end do
for i=k+1 to n do
        a_ik ← l_ik u_kk + Σ_{s=1}^{k-1} l_is u_sk
end do
end do
output (a_kk), (a_kj), (a_ik)
```

We next show how the original data is recovered using Equation (3), (4), and (5) using the same data of Example 1.

Example 2: We obtain elements of L and U after reading the data stored in Node1 and Node2, respectively. We get the original data using Equation (3), (4), and (5).

$$a_{11} = l_{11}u_{11} = 2,\ a_{22} = l_{22}u_{22} + \sum_{s=1}^{1} l_{2s}u_{s2} = l_{22}u_{22} + l_{21}u_{12} = 1$$

$$a_{33} = l_{33}u_{33} + \sum_{s=1}^{2} l_{3s}u_{s3} = l_{33}u_{33} + l_{31}u_{13} + l_{32}u_{23} = 1$$

$$a_{12} = l_{12}u_{22} + \sum_{s=1}^{1} l_{1s}u_{s2} = l_{12}u_{22} + l_{11}u_{12} = 1,\ a_{13} = l_{13}u_{33} + \sum_{s=1}^{2} l_{1s}u_{s3} = l_{13}u_{33} + l_{11}u_{13} + l_{12}u_{23} = 1$$

$$a_{21} = l_{22}u_{21} + \sum_{s=1}^{1} l_{2s}u_{s1} = l_{22}u_{21} + l_{21}u_{11} = 4,\ a_{23} = l_{23}u_{33} + \sum_{s=1}^{2} l_{2s}u_{s3} = l_{23}u_{33} + l_{21}u_{13} + l_{22}u_{23} = 0$$

$$a_{31} = l_{33}u_{31} + \sum_{s=1}^{2} l_{3s}u_{s1} = l_{33}u_{31} + l_{31}u_{11} + l_{32}u_{21} = -2,\ a_{32} = l_{33}u_{32} + \sum_{s=1}^{2} l_{3s}u_{s2} = l_{33}u_{32} + l_{31}u_{12} + l_{32}u_{22} = 2$$

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \begin{bmatrix} 2 & 1 & 1 \\ 4 & 1 & 0 \\ -2 & 2 & 1 \end{bmatrix}$$

4 Performance Evaluation

Availability of a storage node is defined as the probability of the node to be able to service the requests. A common way of modeling the availability of a given set of storage nodes usually assumes the following:

- The failures of the storage nodes are independent.
- The storage nodes have identical availabilities.

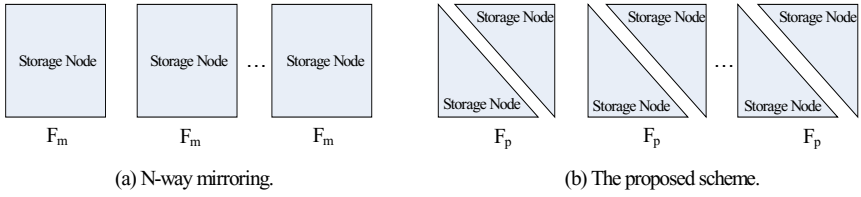


Fig. 1. The structure of mirroring and the proposed scheme.

Fig. 1 shows the structure of mirroring and the proposed scheme. Assume that F_m and F_p is the probability that a node fails in the mirroring and the proposed scheme, respectively, and n is the number of nodes. Note here that a node in the proposed scheme consists of actually a pair of nodes whose size is about half of a node in the mirroring. F_p is the failure probability of one node of the pair node, and thus it is

much smaller than F_m. Here we assume that F_p is half of F_m. Since failure rate of a system grows exponentially as the physical size increases, such assumption will allow very conservative comparison of the availability of the proposed scheme with mirroring.

For mirroring with *n* nodes, the read availability is the probability that at least one of the *n* nodes is good. Thus, the availability is

$$\text{Availability}_{\text{read}}(\text{mirroring}) = 1 - F_m^n \tag{6}$$

The read availability of the proposed scheme is obtained as follows. The probability that one pair node is good is $(1-F_p)^2$. The probability that all the pair nodes fail is $(1-(1-F_p)^2)^n$. The availability is the probability that at least one of the pair nodes is good. Therefore, it is

$$\text{Availability}_{\text{read}}(\text{proposed}) = 1 - (1-(1-F_p)^2)^n \tag{7}$$

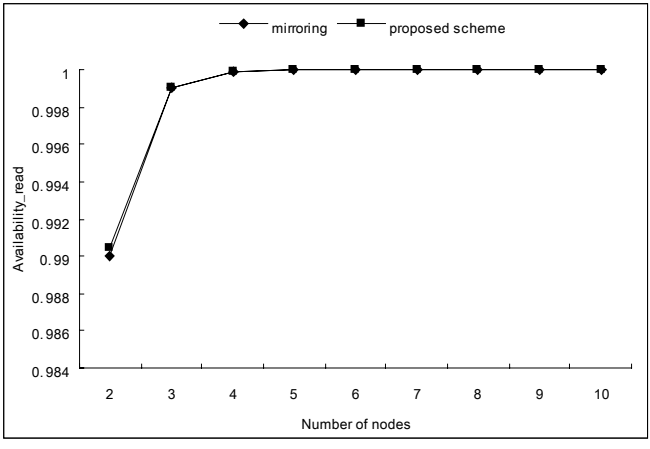


Fig. 2. Comparison of read availabilities.

Fig. 2 shows the comparison of read availability of the proposed scheme along with mirroring scheme. Here up to 10 nodes were tested and F_m is assumed to be 0.1. The figure reveals that the proposed scheme displays slightly higher read availability than mirroring scheme while the availability gets bigger as the number of nodes grows as expected.

We next model the write availability. For write operation, unlike read operation, all the nodes involved must be good. Therefore, for mirroring,

$$\text{Availability}_{\text{write}}(\text{mirroring}) = (1-F_m)^n \tag{8}$$

For the proposed scheme, $(1-F_p)^2$ is the probability that a pair node is good. Therefore,

$$\text{Availability}_{\text{write}}(\text{proposed}) = ((1-F_p)^2)^n \tag{9}$$

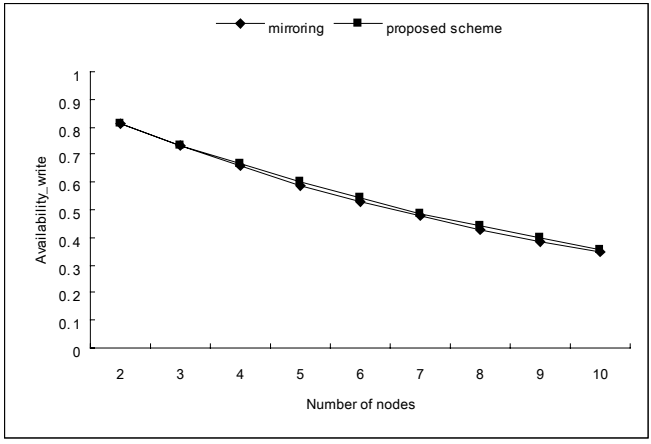


Fig. 3. Comparison of write availabilities.

Fig. 3 compares the write availabilities. Notice that almost the same result as the comparison of the read availabilities is obtained, and the proposed scheme displays slightly higher availability. Recall that F_p was assumed very conservationly. If a more realistic value was taken, the superiority of the proposed scheme will be higher. We also tested a smaller value of node failure probability as $F_m = 0.05$, and still the some results were obtained.

5 Conclusion and Future Work

In this paper we have presented a new data dispersal/encryption scheme. It allows both dispersal and encryption of data at the same time using LU decomposition of matrix. As a result, one cannot extract original data even though the stored data, L or U, matrix are available since it is an NP-hard problem to extract the original matrix using the decomposed matrix data. Therefore, the proposed scheme can provide high security which mirroring scheme cannot. Analysis shows that it allows the same degree of availability compared with mirroring scheme for both read and write operation. Also the proposed scheme needs almost the same storage space as mirroring. A new model considering not only availability but also security in a more formal way will be developed.

References

1. Hui-I. Hsiao and D. J. DeWitt.: A performance study of three high availability data replication strategies: In Proceedings of ICPDIS, (1991) 18-28
2. Darrell D. E. Long.: A Technique for Managing Mirrored Disks: IEEE (2001) 272-277
3. Darrell D. E. Long.: The Management of Replication in a Distributed System: University of California at San Diego, (1988)
4. Mehmet Bakkaloglu, Jay J. Wylie, Chenxi Wang, Gregory R. Ganger.: On Correlated Failures in Survivable Storage Systems: School of Computer Science Carnegie Mellon University, Pittsburgh, PA15213 (2002)
5. R. Cannetti, R. Gennaro, S. Jarecki, H. Krawcxyk and T. Rabin.: Adaptive Security for Threshold Cryptosystems: In Advances in Cryptology-Crypto '99, LNCS, Spriger (1999) 98-115
6. G. R. Blakley, Catherine Meadows.: Security of ramp schemes: Advances in Cryptology, Springer-Verlag, (1985) 242-268
7. Jai Menon, Jeff Riegel and James C. Wyllie.: Algorithms for Software and Low-Cost Hardware RAIDs: COMPCON, 411-418, 1995
8. David A. Patterson, Garth A. Gibson and Randy H. Katz.: A Case for Redundant Arrays of Inexpensive Disks (RAID): In Proceeding of SIGMOD Con- ference, 109-116, 1988
9. A. De Santis and B. Masucci.: Multiple Ramp Schemes: IEEE Trans. Information Theory (1999) 1720-1728
10. E. Karnin, J. Greene, M. Hellman.: On Secret Sharing Systems: IEEE Trans. Information Theory (1983) 35-41
11. Hugo Krawczyk.: Distributed fingerprints and secure information dispersal: ACM Symposium on Principles of Distributed Computing (1993) 207-218
12. Miguel Castro, Barbara Liskov.: Practical Byzantine fault tolerance: Symposium on Operating Systems Design and Implementation, ACM Press (1999) 173-186
13. A. Iyengar et al.: Design and Implementation of a Secure Distributed Computing Systems: Proc. 14th IFIP International Information Security Conference. (SEC 98), ACM Press, New York, 1998
14. Yves Deswarte, L. Blain, Jean-Charles Fabre.: Intrusion tolerance in distributed computing systems: IEEE Symposium on Security and Privacy (1991) 110-121
15. Birkhauser.: Linear Algebra, Birkhauser Boston (1997) 33-37
16. George Nakos, David Joyner.: Linear Algebra with Applications, Brooks/Cole USA (1998) 188-194

An Efficient Conference Key Distribution System Based on Symmetric Balanced Incomplete Block Design

Youngjoo Cho, Changkyun Chi, and Ilyong Chung*

Dept. of Computer Engineering, Chosun University, Kwangju, Korea
iyc@mail.chosun.ac.kr

Abstract. A conference key distribution system is a scheme to generate a conference key, and then to distribute this key to only participants attending at the conference in order to communicate with each other securely. In this paper, an efficient conference key distribution system is presented by employing a symmetric balanced incomplete block design(SBIBD), one class of block designs. Through techniques for creating a conference key and for performing authentication based on identification information, the communication protocol is designed. The protocol presented minimizes the message overhead for generating a conference key. In a special class of SBIBD the message overhead is $O(v\sqrt{v})$, where v is the number of participants. The security of the protocol, which is a significant problem in the construction of secure system, can be proved as computationally difficult to calculate as factoring and discrete logarithms.

1 Introduction

A conference key distribution system(CKDS)[1] is a scheme to generate a common secret key called as a conference key, and to distribute this key to all participants attending at the conference in order to communicate with each other securely. In this paper, identity-based conference key distribution system(CKDS) is presented, in which messages among users are authenticated using each user's identification information. To do authentication[2] is the most important of the security services, because all other security services depend upon it. It is the means of gaining confidence that people or things are who or what they claim to be. An important CKDS system considering authentication was proposed by Shamir[3], where he utilizes ID-based public key system. User's public key contains user's name and address. Shamir and Fiat[4] suggested an authentication mechanism employing discrete logarithm. Okamoto[5] proposed identity-based key distribution system. Ingemarssory, Tang and Wang[6] presented a CKDS on ring network. Koyama and Ohta[7] proposed Identity-based CKDS(ICKDS)

* Corresponding Author: Ilyong Chung (iyc@chosun.ac.kr)

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 647–654, 2004.
© Springer-Verlag Berlin Heidelberg 2004

on ring network, complete graph and star network. Shimbo and Kawamura[8] analyzed several CKDS's.

In case that ICKDS is performed on complete graph. In order for all participants(users) to communicate mutually, a conference key should be generated. We assume that each user has his own key and a conference key is designed by using these keys. One possible manner in which this generation may be carried out is by requiring each user to send its own key to every other user. The relevant computation may then be performed at every site. This method requires $v \times (v-1)$ messages[6] (where v is the number of users in the network) to be sent and one round of message exchange. The conference key is computed as $r_1 \times r_2 \times \ldots \times r_v$, where r_i is user i's secret key. However, as v increases, the message overhead requires $O(v^2)$ and it causes the conference to be delayed.

In this paper, we present efficient conference key distribution system. To accomplish this, a symmetric balanced incomplete block design, one class of block designs[9], is applied for generating the conference key and then this key is distributed to participants. Through this technique for creating a conference key and for performing mutual authentication based on identification information, the communication protocol is designed. The protocol presented minimizes the message overhead for generating a conference key. In a special class of SBIBD the overhead is $O(v\sqrt{v})$, but needs two rounds of message exchange, where v is the number of participants. T he security of the mechanism, which is a significant problem in the construction of secure system, can be proved as computationally difficult to calculate as factoring and discrete logarithms.

This paper is organized as follows. In the next section, we introduce a block design and state the theorems necessary for our presentation. The communication protocol that generates a conference key based on symmetric balanced incomplete block design and distributes all the users is discussed in Section III. The communication scheme considers the sites of distributed systems as constituting the blocks in a block design. This paper concludes with Section IV.

2 Block Design

In this paper, codewords are generated by employing a block design among methods of generation of error-correcting code. By a block design we mean a selection of the subsets of a given set such that some prescribed conditions are satisfied. In some designs, the elements in each of the subsets are also to be ordered in a certain way. A balanced incomplete block design(BIBD) is defined below.

Definition 1. Let $X = \{x_1, x_2, \ldots, x_v\}$ be a set of v objects. A balanced incomplete block design of X is a collection of b k-subsets of X (the k-subsets denoted by $B_1, B_2, \ldots, B_b$) such that the following conditions are satisfied:

1. Each object appears in exactly r of the b blocks.
2. Every two objects appears simultaneously in exactly λ of the b blocks.
3. $k < v$.

Example 1. if $B_1 = \{x_1, x_2, x_3\}$, $B_2 = \{x_4, x_5, x_6\}$, $B_3 = \{x_7, x_8, x_9\}$, $B_4 = \{x_1, x_4, x_7\}$, $B_5 = \{x_2, x_5, x_8\}$, $B_6 = \{x_3, x_6, x_9\}$, $B_7 = \{x_1, x_5, x_9\}$, $B_8 = \{x_2, x_6, x_7\}$, $B_9 = \{x_3, x_4, x_8\}$, $B_{10} = \{x_1, x_6, x_8\}$, $B_{11} = \{x_2, x_4, x_9\}$, $B_{12} = \{x_3, x_5, x_7\}$, then X= $\{x_1, x_2, \ldots x_9\}$, b=12, v=9, r=4, k=3, λ=1. Since a BIBD is characterized by the five parameters b,v,r,k and λ, it is called a (b,v,r,k,λ)-configuration. It is clear that all five of the parameters are not independent. In other words, it is not true that there exists a BIBD for any arbitrary set of these parameters. However, there is no known sufficient condition on the existence of a certain (b,v,r,k,λ)-configuration. We shall show some relations among the parameters that are necessary conditions for the existence of a corresponding (b,v,r,k,λ)-configuration. According the theorem proven in [9], $bk = vr$ and $r(k-1) = \lambda(v-1)$ in a BIBD.

Instead of a list of the k-subsets, a BIBD can be described by the incidence matrix Q, which is a (v × b) matrix with 0ś and 1ś as entries. The rows and columns of the matrix correspond to the blocks and the objects, respectively. The entry in the ith row and the jth column of Q is a 1 if the block B_i contains the object x_j and is a 0 otherwise. The incidence matrix of the BIBD in previous example is described below.

$$\mathrm{Q} = \begin{pmatrix} 1&1&1&0&0&0&0&0&0 \\ 0&0&0&1&1&1&0&0&0 \\ 0&0&0&0&0&0&1&1&1 \\ 1&0&0&1&0&0&1&0&0 \\ 0&1&0&0&1&0&0&1&0 \\ 0&0&1&0&0&1&0&0&1 \\ 1&0&0&0&1&0&0&0&1 \\ 0&1&0&0&0&1&1&0&0 \\ 0&0&1&1&0&0&0&1&0 \\ 1&0&0&0&0&1&0&1&0 \\ 0&1&0&1&0&0&0&0&1 \\ 0&0&1&0&1&0&1&0&0 \end{pmatrix}$$

Fig. 1. (12 x 9) incidence matrix

In some case of a balanced incomplete block design, the number of blocks is the same as that of objects. A balanced incomplete block design is said to be a symmetric balanced incomplete block design(SBIBD) if b=v and r=k. Theorem 1 gives, without the proof, necessary conditions for the existence of SBIBD with given parameter sets (v,k,λ) described in [14].

Theorem 1. Suppose there exists a SBIBD. Put n=k-λ. Then (i) if v is even then n is a square; (ii) if v is odd, then the equation

$$z^2 = nx^2 + (-1)^{(v-1)/2}\lambda y^2 \tag{1}$$

has a solution in integers (x, y, z), not all zero.

Arbitrary two blocks in a SBIDB contains common λ elements. It can be also represented as (v,k,λ)-configuration. In case that $B_1 = \{x_1, x_2, x_4, x_7, x_{11}\}$, $B_2 = \{x_1, x_2, x_3, x_5, x_8\}$, $B_3 = \{x_2, x_3, x_4, x_6, x_9\}$, $B_4 = \{x_3, x_4, x_5, x_7, x_{10}\}$, $B_5 = \{x_4, x_5, x_6, x_8, x_{11}\}$, $B_6 = \{x_5, x_6, x_7, x_9, x_1\}$, $B_7 = \{x_6, x_7, x_8, x_{10}, x_2\}$, $B_8 = \{x_7, x_8, x_9, x_{11}, x_3\}$, $B_9 = \{x_8, x_9, x_{10}, x_1, x_4\}$, $B_{10} = \{x_9, x_{10}, x_{11}, x_2, x_5\}$, $B_{11} = \{x_{10}, x_{11}, x_1, x_3, x_6\}$. Then it becomes (11,5,2)-configuration. A BIBD can be easily derived from the corresponding SBIBD through the intersection of two blocks (B_1, B_i) or the difference of two blocks (B_1, B_i).

Even if a symmetric balanced incomplete block design exists only for certain values of v, normalized Hadamard matrix is utilized for constructing this design of $v = 4n - 1$. Especially, the protocol requires only $O(v\sqrt{v})$ messages based on finite projective planes, which leads to $(k(k-1)+1, k, 1)$-configuration[12].

3 The Design of a Conference Key Distribution System Based on Symmetric Balanced Incomplete Block Design

In order for v participants to communicate mutually, the conference key should be created by utilizing their own keys. The minimal message transmission overhead for this process must be guaranteed. In this paper, the ionic property of error-correcting code is applied and the minimal message overhead requisite to generate this key is maintained. The error-correcting coding method finds out a coset the codeword belongs to, and takes the original value even if a codeword has some errors that can be recoverable. We now apply this concept to the decentralized routing algorithm. Block i and object j correspond to participant i and key j, respectively and the number of blocks is the same as that of participants. For example, seven users take part in conference and each has his own secret key. Each participant at conference computes a conference key based on $(7, 4, 2)$-configuration. (7×7) incidence matrix is now designed below.

$$Q = \begin{pmatrix} 1\,1\,0\,1\,0\,0\,1 \\ 1\,1\,1\,0\,1\,0\,0 \\ 0\,1\,1\,1\,0\,1\,0 \\ 0\,0\,1\,1\,1\,0\,1 \\ 1\,0\,0\,1\,1\,1\,0 \\ 0\,1\,0\,0\,1\,1\,1 \\ 1\,0\,1\,0\,0\,1\,1 \end{pmatrix}$$

Fig. 2. (7×7) incidence matrix

In order to generate a conference key, each receives some keys from users chosen by employing the structure of this matrix. In this paper, two steps are required to calculate the key. User i receives key r_j from user j in case of $Q_{ij} = 1$. We now describe this process from the viewpoint of user 1. First, user 1 receives keys r_2, r_4, r_7 and then make $k_{11} = r_2 \times r_4 \times r_7$, $k_{12} = r_1 \times r_4 \times r_7$, $k_{14} = r_1 \times r_2 \times r_7$, $k_{17} = r_1 \times r_2 \times r_4$, where k_{1j} is the product of $r_{á}s$, a$\in \{1, 2, 4, 7\} - \{j\}$. Simultaneously, other users do the same process. Next, user i receives k_{j1} from user j, if $Q_{j1} = 1$. User 1 receives k_{21}, k_{51}, k_{71} from users 2,5,7. Then the conference key K is calculated as $r_1^2 \times (k_{11} \times k_{21} \times k_{51} \times k_{71})$.

Theorem 2. itshape For user i, the conference key based on (v,k,λ)- configuration is computed as below.

$$K = r_i^{\lambda} \times (\prod_{Q_{ji}=1} k_{ji}) \tag{2}$$

Proof. According to the definition of (v,k,λ)-configuration, each row of (v × v) incidence matrix consists of k 1$\acute{s}$, as does each column. In order for all users to communicate mutually, the conference key should be composed of these secret keys $r_1, r_2, \ldots, r_v$. This key can be obtained by performing the following two steps. On the first step, user i receives (k-1) keys and computes k products, each of which is composed of (k-1) distinguished keys. On the second step, user i receives (k-1) products consisting of (k-1) keys again and collects k_{ji}. Then, the number of keys containing in collected products is $k(k-1)$. Applying $k(k-1) = \lambda(v-1), k(k-1)$ keys are composed of λ $r_{\acute{j}}s$ except his own secret key r_i. Therefore, user i can obtain a conference key by multiplying the product of $(k^2 - k)$ keys by r_i^{λ}.

The sequence of processes for calculating the conference key based on (7,4,2)-configuration is shown (Fig. 3).

User ID	Step 1	Step 2
1	$k_{11} = r_2 \times r_4 \times r_7$, $k_{12} = r_1 \times r_4 \times r_7$ $k_{14} = r_1 \times r_2 \times r_7$, $k_{17} = r_1 \times r_2 \times r_4$	$r_1^2 \times (k_{11} \times k_{21} \times k_{51} \times k_{71})$
2	$k_{22} = r_1 \times r_3 \times r_5$, $k_{21} = r_2 \times r_3 \times r_5$ $k_{23} = r_2 \times r_5 \times r_1$, $k_{25} = r_2 \times r_3 \times r_1$	$r_2^2 \times (k_{22} \times k_{12} \times k_{32} \times k_{62})$
3	$k_{33} = r_2 \times r_4 \times r_6$, $k_{34} = r_2 \times r_3 \times r_6$ $k_{36} = r_2 \times r_4 \times r_3$, $k_{32} = r_3 \times r_4 \times r_6$	$r_3^2 \times (k_{33} \times k_{23} \times k_{43} \times k_{73})$
4	$k_{44} = r_3 \times r_5 \times r_7$, $k_{45} = r_3 \times r_4 \times r_7$ $k_{47} = r_3 \times r_5 \times r_4$, $k_{43} = r_4 \times r_5 \times r_7$	$r_4^2 \times (k_{44} \times k_{14} \times k_{34} \times k_{54})$
5	$k_{55} = r_4 \times r_6 \times r_1$, $k_{56} = r_4 \times r_5 \times r_1$ $k_{51} = r_4 \times r_6 \times r_5$, $k_{54} = r_5 \times r_6 \times r_1$	$r_5^2 \times (k_{55} \times k_{25} \times k_{45} \times k_{65})$
6	$k_{66} = r_2 \times r_5 \times r_7$, $k_{67} = r_2 \times r_5 \times r_6$ $k_{62} = r_6 \times r_5 \times r_7$, $k_{65} = r_2 \times r_6 \times r_7$	$r_6^2 \times (k_{66} \times k_{36} \times k_{56} \times k_{76})$
7	$k_{77} = r_1 \times r_3 \times r_6$, $k_{71} = r_7 \times r_3 \times r_6$ $k_{73} = r_1 \times r_7 \times r_6$, $k_{76} = r_1 \times r_3 \times r_7$	$r_7^2 \times (k_{77} \times k_{17} \times k_{47} \times k_{67})$

Fig. 3. Two steps for designing a conference key based on (7,4,2)-configuration

3.1 The Design of a Conference Key Distribution System Providing Authentication Service

Even a conference key is constructed, we can not guarantee whether the key received from other user is right, which is needed for generating a conference key. To solve this problem, we utilizes user's identity information for authentication. Then a system in the network performs the following steps for creating a secret information.

(1) A system chooses p,q and computes $n = p \times q$, where p,q are primes and approximately 100 digits each.
(2) A relatively large integer e is selected so that e is relatively prime to $(p-1) \times (q-1)$ and d is calculated below

$$e \times d \equiv 1 \bmod (p-1) \times (q-1)$$

(3) Obtain g, which belongs to $GF(p)$ and $GF(q)$.
(4) Compute secret information S_i by employing user i's information ID_i.

$$S_i = ID_i^d$$

A system distributes (e_i, g, n) to all users and user i keeps (d_i, S_i) secret. In order to authenticate user entity and to generate a conference key, we define some notations. $(i \rightarrow j : M)$ indicates that user i transmits information M to user j. (i :) describes that user i stays at his site and does verification or computation. We now present the communication protocol below.

1. $i \rightarrow j : (ID_i, (X_i)^{e_j}, Y_i, t_i)$
   $X_i = g^{e \times r_i} \bmod$ n, $Y_i = S_i \times g^{C_{i_1} \times r_i} \bmod$ n, where $C_i = h(X_i, t_i)$ and $j \in B_i$
   User i belonging to block j creates two information X_i and Y_i for a uthentication, encrypts X_i with e_j and send $(ID_i, (X_i)^{e_j}, Y_i, t_i)$ to user j, where r_i is a secret key of user i and h is a hashing function all the users take in common.

2. j : $X_i = ((X_i)^{e_j})^{d_j}, ID_i = Y_i^e / X_i^{C_{i_2}}$, where $C_{i_2} = h(X_i, t_i)$
   X_i is obtained by decryting $(X_i)^{e_j}$ with d_j. Employing a hashing function and information received from user i, user j can authenticate counterpart's entity. If $ID_i = Y_i^e / X_i^{C_{i_2}}$, then the claim is legitimate.

3. $j \rightarrow p : (ID_j, (X_{jp})^{e_p}, Y_{jp}, t_j)$
   $X_{jp} = X_{P_1} \times X_{P_2} \cdots X_{P_{k-1}}$ where $p_i \in B_j - p$ $Y_{jp} = S_j \times g^{C_{j_1} \times r_j} \bmod$ n, where $C_{j_1} = h(X_{jp}, t_j)$ User j receives (k-1) keys transmitted from users belonging to block j. Then computes X_{jp} and Y_{jp}, and then send $(ID_j, (X_{jp})^{e_p}, Y_{jp}, t_j)$ to user p.

4. p : $X_{jp} = ((X_{jp})^{e_p})^{d_p}, ID_j = Y_{jp}^e / X_{jp}^{C_{j_2}}$, where $C_{j_2} = h(X_{jp}, t_j)$
   X_{jp} can be computed when $(X_{jp})^{e_p}$ is decrypted with d_p . Then user p

authenticates user j's entity by using information obtained from user j. In case that $ID_j = Y_{jp}^e / X_{jp}^{C_{j_2}}$, authentication process is succeeded.

Theorem 2. If $ID_i = Y_i^e / X_i^{C_i}$, then user j gains confidence that information for generating a conference key is transmitted from user i.

Proof. $Y_i^e / X_i^{C_{i_1}} = (S_i \times g^{C_{i_1} \times r_i})^e / (g^{e \times r_i})^{C_{i_2}} \equiv S_i^e$, if $C_{i_1} = C_{i_2}$ Since $S_i = ID_i^d$, $(ID_i^d)^e$ is ID_i by Euler's Theorem.

In order to compute a conference key, user p utilizes his own secret key and X_{jp}'s transmitted from the users in block p. Since each secret key and e appear λ times and $\lambda(v-1)$ times in X_{jp}'s, respectively. Then, user p calculates a conference key below.

$$K = (X_{jp_1} \times X_{jp_2} \cdots \times X_{jp_{k-1}}) \times g^{e_p^\lambda \times r_p^\lambda}$$

Looking into the expression above, if λ is small, this scheme will be better since the time complexity for the computation of conference key should be reduced.

3.2 Analysis of the Proposed Conference Key Distribution System

The communication protocol based on (v, k, λ)-configuration is now analyzed. Since the first and second steps require $v \times (k-1)$ messages each, the complexity is $O(v \times k)$ by Theorem 2. According to $k(k-1) = \lambda(v-1)$, k is determined by the values of v and λ. In case of $\lambda = 1$, k becomes approximately $\sqrt{v}$. So, the complexity is $O(v\sqrt{v})$. Security of the protocol is now considered. In order to reveal secret information S_i, given e and n, d can not be computed since no polynomial algorithm has been found for solving factorization problem. The secret key r_i should be protected. Given X_i, to get r_i is a difficult problem because finding discrete logarithm is generally a hard problem. Therefore, security of the communication protocol is computationally secure.

4 Conclusion

An efficient identity-based conference key distribution system is developed for group communication service, on which only participants in group communicate each other. To accomplish this, (v,k,λ)-configuration method is applied for generating a conference key and then this key is distributed to participants through authentication technique. The communication protocol requires two rounds of message exchange and $O(v\sqrt{v})$ messages in case of λ=1, compared with $O(v^2)$ messages needed for one round of message exchange.

The security of the protocol is a significant problem in the construction of secure system. In this paper, it can be proved as computationally difficult to calculate as factoring and discrete logarithms.

References

1. C. Chang, T. Wu and C. Chen, *The Design of a Conference Key Distribution System,* Proc. of ASIACRYPT'92, pp. 11.1-11.6, 1992.
2. J. Seberry and J. Pieprzyk, Cryptography: An Introduction of Computer Security. Prentice-Hall, New York, 1988.
3. A. Shamir *Identity-based cryptosystems and signature schemes*, Proc. of Crypto'84, Lecture Notes in Computer Science no. 196, Springer-Verlag, pp.47 53, 1985.
4. A. Fiat and A. Shamir, *How to prove yourself: Practical solutions to identification and signature schemes*, Proc. of Crypto'86, Lecture Notes in Computer Science no. 263, Springer-Verlag, pp. 186 194, 1987.
5. T. Okamoto, *Proposal for identity-based key distribution system*, Electron. Lett., no. 22, pp. 1283 1284, 1986.
6. I. Ingemarsson, D. T. Tang and C. K. Wong, *A Conference Key Distribution system*, IEEE Trans. on Info. Theory vol. IT-28, pp.714 720, 1982.
7. K. Koyama and K. Ohta, *Identity-Based Conference Key Distribution System*, Proc. of Crypto'87, Lecture Notes In Computer Science, 1987.
8. A. Shimbo and S. Kawamura, *Cryptoanalysis of several Conference Key Distribution Schemes,* Proc. of ASIACRYPT'91, pp. 155-160, 1991.
9. C. Liu, *Block Designs* in Introduction to Combinatorial Mathematics, McGraw-Hill, New York, 1968.
10. M. Rhee, error Correcting Coding Theory, McGraw-Hill, New York, 1989.
11. D. Welsh, Codes and Cryptography, Oxford Science Pub., Oxford, 1988.
12. J. Ryou, *A Load Balancing Algorithm in Distributed Computing Systems,* J. of Korea Info. Sci. Soc., Vol.20, No.3, pp. 430-441, 1993.
13. T. Lee and I. Chung, *The Design of Authentication Mechanism Employing the Block Design for Information Security in CORBA Environment,* J. Korean Inst. of Commun. Sci., vol. 24, no. 3B, pp. 330-337, Mar. 1999.
14. P.J. Cameron and J.H. Van Lint, Graph Theory, Coding Theory and Block Design, Cambridge University Press, p.4, 1975

Multiparty Key Agreement Protocol with Cheater Identification Based on Shamir Secret Sharing

Kee-Young Yoo, Eun-Kyung Ryu, and Jae-Yuel Im

Department of Computer Engineering, Kyungpook National University,
Daegu 702-701, Republic of Korea
yook@knu.ac.kr

Abstract. Recently, multiparty key agreement protocols with cheater identification based on Shamir secret sharing was presented by Pieprzyk and Li and Y. M. Tseng, respectively. However, in their multiparty key agreement protocols there are mistakes in computing a common secret key. Also Tseng's cheater identification scheme cannot identify cheaters perfectly. In this paper, we present a concrete multiparty key agreement protocol with more efficient and perfect cheater identification than Tseng's protocol in computations and communications. Performance comparison and security analysis are given in this paper.

1 Introduction

A multiparty key agreement is one of the fundamental cryptographic primitives. This is required in situation where two or more principals want to communicate securely among themselves over an open distributed network or ubiquitous computing environment using shared secret key. The key agreement protocol is that all principals interact with each other and then compute a common secret key. Note that the common secret key is collectively determined by all principals in the key agreement protocol. The latter one is more suitable for the distributed ad-hoc network environment and ubiquitous computing environment [5,6].

There have been intensive researches on multiparty key agreement protocols [1,3,7] . The first multiparty key agreement was proposed by Ingemasson et al. [8] which is a generalization of the Diffie-Hellman protocol [4] since multiparty key agreement can be seen as a generalization of two-party key agreement. Burmester and Desmedt [2] showed that if participants are able to structure itself into a ring, then after each principal broadcasts to two neighbors, all participants are able to generate a common secret key. Just and Vaudenay [9] presented a multiparty key agreement protocol in which two-party authentication is extended into the group authentication.

Almost all previous multi-party key agreement protocols are based on Diffie-Hellman key agreement. However, some researchers have exploited secret sharing scheme in multi-party key agreement since it allows a group of users to cooperate to derive a secret value. Pieprzyk and Li [10] presented multiparty key agreement

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 655–664, 2004.
© Springer-Verlag Berlin Heidelberg 2004

protocols based on Shamir [11] secret sharing. The protocols can achieve the intended security goals and can detect there exist fraudulent principals, so called cheaters. But, any principal as well as the trusted registry can not identify the cheaters. Recently, Y. M. Tseng [12] appended a cheater identification scheme that allows the trusted registry to identify the fraudulent principals into the Pieprzyk-Li's [10] multiparty key agreement protocols. His modified protocols also achieve the same security goals as the Pieprzyk-Li's protocol. However, each principal in the Tseng's protocol generates two more polynomial values used for cheater identification. Although his protocol used $2n$ more polynomial values, it still can not identify cheaters perfectly.

In this paper, we propose a multiparty key agreement protocol with an efficient cheater identification based on Shamir secret sharing. Our protocol detects there exists a cheater and identifies the cheater. Moreover, in the protocol each principal generates less polynomial values than Tseng's protocol for cheater identification. We show that computational and communication efficiency of our protocol are better than that of Tseng's protocol.

The remainder of this paper is organized as follows. In the next section, we review briefly Pieprzyk-Li's and Tseng's protocols and discuss the security weakness on Tseng's protocols. The proposed protocol is described in Section 3. In Section 4, we discuss performance comparison and security analysis. Section 5 gives conclusions.

2 Reviews of Related Previous Works

In this section, we review the Pieprzyk-Li's [10] and Tseng's [12] multiparty key agreement protocol. Also we discuss some mistakes and weakness of Tseng's multi-part key agreement protocol with cheater identification.

2.1 Review of the Pieprzyk and Li's Multiparty Key Agreement Protocol

The Pieprzyk-Li's multiparty key agreement applies secret sharing generated independently by principals. Assume that there are n principals who are eligible to participate in a conference. Each principal plays the role of dealer, i.e. each principal sets up his/her own $2n$ shares $(z_k, f_i(z_k))$ defined by polynomial $f_i(z)$ of degree n and sends securely and broadcasts shares to the others of the group through prearranged secure and authenticated channel. To recovery the secret, each principal combines shares sent from the others with her own share to compute the secret, i.e. a multiparty key that will be shared by all principals, using the Lagrange polynomial interpolation of the polynomial $F(z) = \sum_{i=1}^{n} f_i(z)$ In Pieprzyk and Li's paper, two protocols, i.e. main and threshold protocol are mentioned. In this paper, we have an interest in main protocol in which the access structure consists of all principals participated in a conference. In [10], it is proved that their protocol achieves security goals: key freshness, key confidentiality, group authentication, key confirmation.

2.2 Tseng's Multiparty Key Agreement Protocol

Tseng modified the Pieprzyk-Li multiparty key agreement protocol in order to append the cheater identification phase [12]. The Tseng's protocol is divided into four phases: the registration phase, the initialization phase, the key agreement phase, and the cheater identification phase.

During the registration phase, the trusted registry R chooses and publishes two large prime numbers p and q such that q divides $p-1$. Let g be a generator with order q in Z_q^*. In addition, R also selects a random integer $r \in Z_q^*$ and publishes $\alpha = g^r \bmod p$ which is used to supply principals with key freshness in each conference. Suppose there are n principals $\{P_1, \ldots, P_n\}$. Each principal P_i $(1 \le i \le n)$ chooses a secret key $x_i \in Z_q^*$ and submits the public key $y_i = g^{x_i} \bmod p$ to R. Afterward, R displays a read-only list of each principal's public key with his/her identifier.

During the initialization phase, the following steps are executed independently by each principal.

Step 1: P_i designs a $(n+1,\ 2n)$ Shamir threshold scheme, in which there are $2n$ shares with threshold $n+1$. Let the scheme be defined by a random polynomial $f_i(z)$ with degree at most n. Suppose that

$$f_i(z) = a_{i,0} + a_{i,1}z + a_{i,2}z^2 + \cdots + a_{i,n}z^n \bmod p \tag{1}$$

where $a_{i,j} \in Z_q^* (j = 1 \ldots n)$ are chosen at random.

Step 2: P_i computes $n+1$ pairs of shares $s_{i,j} = (s_{i,j}^{(1)} = f_i(2j-1), s_{i,j}^{(2)} = f_i(2j))$ for $j = 1 \ldots n{+}1$. Then, P_i computes and publishes $\delta_{i,j} = g^{s_{i,j}^{(2)}} \bmod p$ for $j = 1 \ldots n$. Additionally, P_i also computes and sends $v_{i,1} = g^{s_{i,n+1}^{(1)}} \bmod p$ and $v_{i,2} = g^{s_{i,n+1}^{(2)}} \bmod p$ to the trusted registry R via a secure channel.

Step 3: Finally, P_i sends $s_{i,j}^{(2)}$ to other principals P_j via a secure channel, where $j = 1 \ldots n$ and $j \ne i$.

After performing Step 3, each principal P_i obtains a sequence of n elements $(s_{1,i}^{(2)}, \ldots, s_{n,i}^{(2)})$. Then, P_i computes the secret share $S_i^{(2)} = \sum_{j=1}^{n} s_{j,i}^{(2)}$, that is, $S_i^{(2)} = F(2i)$ where the polynomial $F(z) = \sum_{i=1}^{n} f_i(z) \bmod q$

During the key agreement phase, each principal P_i performs the following steps to compute a common secret key using Lagrange polynomial interpolation.

Step 1: P_i asks and fetches $\alpha = g^r \bmod p$ from the trusted registry R.

Step 2: P_i computes and broadcasts public shares $\beta_{i,j} = \alpha^{s_{i,j}^{(1)}} \bmod p$ for $j = 1 \ldots n$.

Step 3: After obtaining $\beta_{i,j}$ from other principals, P_i computes $\alpha^{S_j^{(1)}} = \alpha^{F(2j-1)} = \alpha^{\sum_{i=1}^{n} s_{i,j}^{(1)}} = \prod_{i=1}^{n} \beta_{i,j} \bmod p$, for $j = 1 \ldots n$.

Step 4: P_i uses his/her secret share $S_i^{(2)}$ and n public shares $\alpha^{S_j^{(1)}} (j = 1 \ldots n)$ to recover the common secret S by adopting the Lagrange interpolation but for exponents as follows.

$$S = \alpha^{F(0)} = (\alpha^{S_i^{(2)}})^b \prod_{j=1}^{n} (\alpha^{S_j^{(1)}})^{b_j} \quad (2)$$

where $b = \prod_{j=2,4,\ldots 2n; j \neq 2i} \frac{j}{j-2i}$ and $b_j = \prod_{j=1,3,\ldots 2n-1; l \neq 2j-1} \frac{l}{l-2j+1}$.

Step 5: P_i takes the common secret S, his/her name id_i and α to compute a checking string $\sigma = H(S||id_i||\alpha)$, where H is a public collision hash function. Then, P_i broadcasts the triplet $(\sigma,\ id_i,\ \alpha)$.

Step 6: After collecting $(\sigma,\ id_j,\ \alpha)$ from other principals, P_i verifies them using his/her own pre-computed secret S. If these checks hold, P_i is ready for the conference. Otherwise, P_i announces the error and aborts the conference.

The cheater identification phase shows how to identify fraudulent principals. When key confirmation is not satisfied at the Step 6 in the key agreement phase, the trusted registry R executes the following steps for each principal P_i.

Step 1: R takes $v_{i,1} = g^{s_{i,n+1}^{(1)}} \bmod p$ and $v_{i,2} = g^{s_{i,n+1}^{(2)}} \bmod p$.

Step 2: Because $\alpha = g^r \bmod p$ and the random integer r is selected by the trusted registry R, he computes $\beta_{i,n+1} = v_{i,1}^r = \alpha^{s_{i,n+1}^{(1)}} \bmod p$ and $\omega = v_{i,2}^r = \alpha^{s_{i,n+1}^{(2)}} \bmod p$.

Step 3: R uses n public values $\beta_{i,j} = \alpha^{s_{i,j}^{(1)}} \bmod p$, $(j = 1...n)$ published by P_i and $\beta_{i,n+1} = v_{i,1}^r = \alpha^{s_{i,n+1}^{(2)}} \bmod p$ to compute the first checking value $C_{i,1} = \alpha^{f_i(0)} \bmod p$. Note that R still use the adopting the Lagrange interpolation but for exponents as follow.

$$C_{i,1} = \alpha^{f_i(0)} = \prod_{j=1}^{n+1} (\alpha^{s_{i,j}^{(1)}})^{b_j} \quad (3)$$

where $b_j = \prod_{j=1,3,\ldots 2n+1; l \neq 2j-1} \frac{l}{l-2j+1}$ are Lagarange coefficients.

Step 4: R also uses n public values $\beta_{i,j} = \alpha^{s_{i,j}^{(1)}} \bmod p, (j = 1 \ldots n)$ and $\omega = v_{i,2}^r = \alpha^{s_{i,n+1}^{(2)}} \bmod p$ to compute the second checking value $C_{i,2} = \alpha^{f_i(0)} \bmod p$ by adopting the Lagrange interpolation.

Step 5: R checks whether $C_{i,1} = C_{i,2}$ holds or not. If it does not hold, then P_i is a fraudulent principal.

2.3 Mistakes and Weakness of the Previous Protocols

Here we describe mistakes in the initialization and key agreement phases of the previous two protocols and weakness in the cheater identification phase of the Tseng's protocol.

In the previous two protocols, the multiplicative group Z_q^* of order q that is a subgroup of Z_p^* is used in computation for a multiparty key where p and q are large primes such that q divides $p-1$. Let $\alpha = g^r \bmod p$ where $r \in Z_q^*$ and g is a generator of Z_q^*. While exponentiation operations must be performed modulo p,

computations of values that are used for exponents must be performed modulo q. Modular arithmetic notations never be denoted in Pieprzyk-Li protocol. But in the Tseng's protocol, modular arithmetic notations are partially used. However, modular arithmetic notations are misused in some steps of Tseng's protocol. For example, see equation (1). In the Step 1 of the initialization phase, each principal generates shares from polynomial $f_i(z) \mod p$. These shares are used as exponents. Thus, evaluation $f_i(z) \mod p$ must be changed into $f_i(z) \mod q$.

In equation (2) in Step 4 of the key agreement phase and equation (3) of the cheater identification, mod q notations are missing in the computation of Lagrange coefficients b and b_j. The mod q notation is required because these are not integer computation. Moreover the equations that compute Lagrange coefficients b and b_j of equation (2) are wrong since the number of terms added is not $n-1$, but n.

Now we discuss about cheater identification. Certainly, cheaters may disturb the protocol, causing the other principals not to hold an exact multiparty key. Although other honest principals can detect there exists a cheater, no one (including the trusted registry R) can identify the cheater in the Pieprzyk-Li protocol. So, Tseng appends cheater identification phase into the Pieprzyk-Li protocol to identify the cheater. But, Tseng's protocol does not identify perfectly the cheaters. Let us consider the following cases: If any principal causes any one of the following three cases, the principals who join in a conference can not recover a common multiparty key.

(1) During the initialization phase, the fraudulent principal, so-called cheater can intentionally select his/her random two different polynomials and send values, $s_{i,j}^{(1)}$ and $s_{i,j}^{(2)}, (j = 1...n)$, generated by the different polynomials, respectively, to any one of the other principals.
(2) During the initialization phase, the cheater can purposely send wrong secret shares, $s_{i,j}^{(2)}, (j = 1...n)$, to any one of the other principals.
(3) During the key agreement phase, the cheater can purposely broadcast wrong public shares, $\alpha^{s_{i,j}^{(1)}}, (1...n)$, to the other principals.

Case (1): During the initialization phase of Tseng's protocol, the trusted registry R knows values $\delta_{i,j} = g^{s_{i,j}^{(2)}} \mod p,\ 1 \le j \le n$ that principal P_i publishes, and values $v_{i,1} = g^{s_{i,n+1}^{(1)}} \mod p$ and $v_{i,2} = g^{s_{i,n+1}^{(2)}} \mod p$ that P_i sends to R. So R can compute $C_{i,1} = g^{f_i(0)} \mod p$ using $\delta_{i,j},\ 1 \le j \le n$ and $v_{i,1}$, and also can compute $C_{i,2} = g^{f_i(0)} \mod p$ using $\delta_{i,j},\ 1 \le j \le n$ and $v_{i,2}$ by Lagrange polynomial interpolation. Tseng proved that if P_i is a honest principal, i.e. $n+2$ values $s_{i,j}^{(2)},\ 1 \le j \le n+1$ and $s_{i,n+1}^{(1)}$ are defined by a polynomial $f_i(z)$, then $C_{i,1} = C_{i,2}$, otherwise P_i is a cheater. However it is not considered whether P_i generates values $s_{i,j}^{(1)}, (j = 1...n)$ by the same polynomial $f_i(z)$. Thus, P_i can determine a different polynomial $f_i'(z)$ of degree n satisfying $f_i'(0) = f_i(0)$, $f_i'(2n+1) = s_{i,n+1}^{(1)} = f_i(2n+1)$, and $f_i'(2(n+1)) = s_{i,n+1}^{(2)} = f_i(2(n+1))$ such that $f_i'(2j-1) = s_{i,j}^{(1)}, (j = 1...n)$. If principal P_i uses two functions $f_i(z)$ and

$f_i'(z)$, then a common multiparty key is not recovered. In this case, the trusted registry R can not identify P_i must be a cheater.

Case (2): During the initialization phase of the Tseng's protocol, if principal P_i sends different values $s_{i,j}^{(2)}$, $j = 1...n$ from those defined by the random polynomial $f_i(z)$ to the other principals, it is impossible to identify a cheater because there are no steps checking the values in the cheater identification phase.

Case (3): The cheater identification phase of Tseng's protocol works well in identifying a cheater like this case.

3 The Proposed Protocol with Cheater Identification

In this section, we present an improved multiparty key agreement protocol with cheater identification based on Shamir secret sharing. The registration phase is the same as that of the previous protocols.

3.1 The Initialization Phase

In this phase, each principle who registers herself with a trusted registry creates her $2n$ shares, and sends secretly one of $2n$ shares to the trusted registry which will be used for cheater identification, and distributes n-1 shares to other principals secretly for making principal's secret share.

Step 1: Each principal P_i builds a random polynomial $f_i(z)$ over Z_q of degree at most n in order to design Shamir secret sharing scheme. Suppose that

$$f_i(z) = a_{i,n}z^n + a_{i,n-1}z^{n-1} + \cdots + a_{i,1}z + a_{i,0} \mod q \qquad (4)$$

where the coefficients $a_{i,j} \in Z_q^*$ are chosen at random for $1 \le j \le n$.

Step 2: Each P_i prepares $2n$ shares $s_{i,j}^{(1)} = f_i(z_{2j-1})$ and $s_{i,j}^{(2)} = f_i(z_{2j})$ for $2n$ public coordinates z_{2j-1} and z_{2j}, $(1 \le j \le n)$. Additionally P_i sends $g^{s_{i,i}^{(2)}} (= g^{f_i(z_{2i})}) \bmod p$ to the trusted registry R via a secure channel. Eventually R keeps secretly them that are used for the cheater identification phase.

Step 3: Each principal P_i sends $s_{i,j}^{(2)} (= f_i(z_{2j}))$ $(1 \le j \le n,\ j \ne i)$ signed by P_i to the other principals P_j via a secure channel. Each principal P_i receives the signed values $s_{j,i}^{(2)}$ from other principals P_j and sends the signed values to the trusted registry R.

Eventually, each principal P_i has a sequence of n elements $(s_{1,i}^{(2)}, \cdots, s_{n,i}^{(2)})$. Then P_i can compute a secret share $SS_j \equiv \sum_{k=1}^{n} s_{k,i}^{(2)} = \sum_{k=1}^{n} f_k(z_{2i}) \mod q$, i. e. $SS_j = F(z_{2i}) \mod q$ where the polynomial $F(z) = \sum_{i=1}^{n} f_i(z) \bmod q$. Also R can collect $s_{i,j}^{(2)} (= f_i(z_{2j}))$ $(1 \le j \le n,\ j \ne i)$ which each principal P_i sends to other principal P_j.

Note that $2n$ public coordinates z_{2j-1} and z_{2j} in Step 2 can be published in the read-only list by the trusted registry.

3.2 The Key Agreement Phase

To broadcast n shares for making n pubic shares and therefore to enable themselves to computer a common secret key, each principal performs the following steps independently.

Step 1: Each principal P_i asks and fetches $\alpha(= g^r)$ from the trusted registry R.

Step 2: Each principal P_i computes and broadcasts public shares $\beta_{i,j} = \alpha^{s_{i,j}^{(1)}} \bmod p$ for $1 \le j \le n$.

Step 3: After obtaining for $\beta_{k,j}$, $(1 \le k, j \le n,\ k \ne i)$ from other principals, P_i computes public shares α^{PS_j}, $(1 \le j \le n)$

$$\alpha^{PS_j} \equiv \alpha^{\sum_{k=1}^{n} s_{k,j}^{(1)} \bmod q} \bmod p = \prod_{k=1}^{n} \alpha^{s_{k,j}^{(1)}} \bmod p = \prod_{k=1}^{n} \beta_{k,j} \bmod p$$

Note that $\alpha^{PS_j} = \alpha^{F(z_{2j-1}) \bmod q} \bmod p$, $(1 \le j \le n)$.

Step 4: Each principal P_i uses his/her secret share SS_i and n public shares α^{PS_j}, $(1 \le j \le n)$ to recover the common secret key $S = \alpha^{F(0)} \bmod p$ by adopting the Lagrange polynomial interpolation as follows:

$$S \equiv \alpha^{F(0)} \bmod p = (\alpha^{SS_i})^{l^i \bmod q} \prod_{j=1}^{n} (\alpha^{PS_j})^{l_j^i \bmod q} \bmod p$$

where $l^i = \prod_{k=1}^{n} (z_{2k-1}) (z_{2k-1} - z_{2i})^{-1} \bmod q$ and

$$l_j^i = z_{2i}(z_{2i} - z_{2j-1})^{-1} \prod_{k=1;k\ne j}^{n} (z_{2k-1})(z_{2k-1} - z_{2j-1})^{-1} \bmod q.$$

Note that Step 5 and 6 are the same as them of the key agreement phase of Tseng's protocol. If all principals are honest and follow the steps of protocol, each principal get the same common secret key $S = \alpha^{F(0)} \bmod p$.

3.3 The Cheater Identification Phase

When all principals do not share the same common secret key, the trusted registry R performs the following steps to identify whether each principal P_i is a cheater.

Step 1: The registry R chooses a random coordinate $z_r \in Z_q^*$ for $z_r \ne z_{2j}$ and $z_r \ne z_{2j-1}$ $(1 \le j \le n)$.

Step 2: The registry R knows n public values $\beta_{i,j} = \alpha^{s_{i,j}^{(1)}} \bmod p = \alpha^{f_i(z_{2j-1})} \bmod p$, $(1 \le j \le n)$ which are broadcasted by P_i and one secret value $\alpha^{s_{i,i}^{(2)}} (= \alpha^{f_i(z_{2i})})$ that R keeps secretly in the initialization phase. The registry R uses them to compute the checking value $C_i^1 \equiv \alpha^{f_i(z_r)} \bmod p$ by adopting the Lagrange interpolation as follows:

$$C_i^1 \equiv \alpha^{f_i(z_r)} \bmod p = (\alpha^{S_{i,i}^{(2)}})^{l^i \bmod q} \prod_{j=1}^{n} (\beta_{i,j})^{l_j^i} \tag{5}$$

where $l^i = \prod_{k=1}^{n} (z_r - z_{2k-1})(z_{2i} - z_{2k-1})^{-1} \bmod q$ and

$$l_j^i = (z_r - z_{2i})(z_{2j-1} - z_{2i})^{-1} \prod_{\substack{k=1 \\ k \neq j}}^{n} (z_r - z_{2k-1})(z_{2j-1} - z_{2k-1})^{-1} \bmod q.$$

Step 3: The registry R selects one public value $\beta_{i,j}$, say $\beta_{i,1} = \alpha^{s_{i,1}^{(1)}} = \alpha^{f_i(z_1)} \bmod p$, out of n public values $\beta_{i,j} = \alpha^{s_{i,j}^{(1)}} \bmod p$ and obtains $n-1$ values $s_{i,j}^{(2)} = f_i(z_{2j})$, $(1 \leq j \leq n, j \neq i)$ which P_i sends to other principals, and $\alpha^{s_{i,i}^{(2)}} (= \alpha^{f_i(z_{2i})})$, since $\alpha^{s_{i,i}^{(2)}} = (g^{s_{i,i}^{(2)}})^r$. Also R uses them to compute the checking value $C_i^2 \equiv \alpha^{f_i(z_r)} \bmod p$ by adapting the Lagrange interpolation as follows:

$$C_i^2 \equiv \alpha^{f_i(z_r)} \bmod p = (\beta_{i,1})^{l^i} \alpha^t \bmod p, t = \sum_{j=1}^{n} s_{i,j}^{(2)} \times l_j^i \bmod q \qquad (6)$$

where $l^i = \prod_{k=1}^{n} (z_r - z_{2k})(z_1 - z_{2k})^{-1} \bmod q$ and

$$l_j^i = (z_r - z_1)(z_{2j} - z_1)^{-1} \prod_{\substack{k=1 \\ k \neq j}}^{n} (z_r - z_{2k})(z_{2j} - z_{2k})^{-1} \bmod q.$$

Step 4: The registry R checks whether $C_i^1 = C_i^2$ holds or not. If it does not hold, then P_i is a cheater.

4 Security Analysis and Performance Comparison

In this section, we discuss security and performance for our modified protocol. Each principal who joins in Tseng's protocol generates $2(n+1)$ shares of his/her random polynomial $f_i(z)$. However in our protocol each principal generates $2n$ shares like Pieprzyk-Li's protocols. Obviously, security for the key agreement of our protocol is based on the same cryptographic assumptions as the Pieprzyk-Li's protocol. The difference between our protocol and Pieprzyk-Li's protocol is the Step 2 and Step 3 in the initialization phase and the cheater identification phase. At Step 2 in the initialization phase, each principal P_i sends $g^{f_i(z_{2i})}$ to the trusted registry R. Any attacker does not know $f_i(z_{2i})$ and $\alpha^{f_i(z_{2i})}$ because of the Discrete Logarithmic Problem (DLP). But only the trusted registry R knows $\alpha^{f_i(z_{2i})}$ because only R knows r and $\alpha = g^r$. At Step 3, each principal P_i sends the signed values which he/she receives signed values $s_{j,i}^{(2)} = f_j(z_{2i})$, $(j = 1...n, j \neq i)$ from other principals P_j to the trusted registry R via a secure channel. Any attacker can know values $\alpha^{f_i(2j-1)}, j = 1...n$ because principal P_i broadcasts public values. However any one except R and P_i never knows such values $f_i(z_{2j})$. Thus, any attacker can not obtain $\alpha^{f_i(0)} \bmod p$ for any i because the degree of each random polynomial $f_i(z)$ is n and $n+1$ points are needed to compute $f_i(0)$. Moreover, the attacker can not obtain a common multiparty key $\alpha^{F(0)} \bmod p$.

When the key agreement protocol fails, the trusted registry R performs the cheater identification phase to check whether principal P_i is a cheater. In the

previous section, we showed weakness of Tseng's cheater identification. Now we show that our cheater identification phase is more efficient and perfect than Tseng's cheater identification. In our cheater identification, R have values $g^{s_{i,i}^{(2)}}$ $(= g^{f_i(z_{2i})})$ and $f_i(z_{2j}), (j = 1...n, j \neq i)$ that P_i sends to R in the initialization phase. Also R can obtain public values $\beta_{i,j} = \alpha^{f_i(2j-1)}, j = 1...n$ published by P_i during the key agreement phase. In Step 2 and 3, R computes $C_i^1 = \alpha^{f_i(z_r)}$ and $C_i^2 = \alpha^{f_i(z_r)}$ for a random coordinate $z_r \in Z_q^*$. If principal P_i is a honest principal, then $C_i^1 = C_i^2$ holds. If P_i gives occasion to any of case (2) and (3) in the subsection 2.3, then R easily identifies the cheater. If P_i guesses exactly z_r that R chooses at random, he/she can make two different polynomials $f_i(z)$ and $f_i'(z)$ of degree n such that $\alpha^{f_i(z_r)} = C_i^1 \equiv C_i^2 = \alpha^{f_i'(z_r)}$. In this case, although the key agreement protocol fails, R cannot identify P_i as a cheater. However, it is probabilistically impossible that P_i guesses exactly z_r and furthermore that P_i finds two polynomials $f_i(z)$ and $f_i'(z)$ meeting at z_r. In spite of the case (1) occurs in our protocol, R can identify cheaters. Thus, we can say that the cheater identification phase is more efficient and perfect than Tseng's phase.

The following Table 1 shows the performance comparison of our protocols with Tseng's protocol.

Table 1. Comparison of Communication and Computation

		Yuh-Min Tseng protocols	Proposed protocols
Communications	Setup	n message	$2n-1$ message
	Key agreement	$2n-2$ message	$2n-2$ message
Computations	Setup	$(2n^2+2n)\mathbf{M}$ $(n+2)\mathbf{E}$	$2n^2\mathbf{M}$ $1\mathbf{E}$
	Key agreement	$(3n^2+n-1)\mathbf{M}$ $(3n+2)\mathbf{E}$	$(3n^2+n-1)\mathbf{M}$ $(3n+2)\mathbf{E}$
	Cheater identification	$(8n^3+8n^2-4n)\mathbf{M}$ $(8n^2+10n)\mathbf{E}$	$(4n^3+4n^2-2n)\mathbf{M}$ $(5n^2+4n)\mathbf{E}$

* M: multiplication, E: exponentiation

The entry equation means communication and computation overhead required by each principal. But computation overhead of cheater identification means operation amount required in checking for all principals. As shown in Table 1, during the setup phase of our protocol, the communication overhead done by each principal is increased two times and the computation overhead is decreased $n+1$ exponentiations. By the way, the setup phase is only performed once so that the variation of overheads does not affect seriously the performance of protocols. As mentioned in subsection 2.3, Tseng's cheater identification phase is not perfect. But our cheater identification is able to identify the cheater for all cases. Moreover computation overhead of our cheater identification phase is much less that that of Tseng's.

5 Conclusions

We have proposed an efficient and perfect multiparty key agreement protocol with cheater identification based on Shamir secret sharing. We have made a slight correction on the Tseng's key agreement protocol and improved the cheater identification of Tseng's protocol that overcomes the weakness of cheater identification. Although Tseng's protocol uses more polynomial values and Lagrange polynomial interpolations, his cheater identification is not perfect. But our protocols can identify the cheater almost perfectly using less polynomial values. Moreover, we have shown that computational and communication efficiency of our protocol is better than that of Tseng's protocol.

Acknowledgements. We would like to thank anonymous reviewers for their helpful comments. This work was supported by the Brain Korea 21 Project in 2003.

References

1. Boyd, C., Mathuria, A.: Protocols for Authentication and Key Establishment. Springer-Verlag, Berlin Heidelberg (2003)
2. Burmester, M., Desmedt, Y.: A secure and efficient conference key distribution system. EUROCRYPT'94 Lecture Notes in Computer Science, Vol. 950. Springer-Verlag, Berlin Heidelberg (1996) 275–286
3. Chang, C. C., Hwang, R. J.: Efficient cheater identification method for threshold schemes. IEE Proc.-Comput. Digit. Tech. Vol. 144 (1) (1997) 23–27
4. Diffie, W., Hellman, M. E.: New directions in cryptography. IEEE Trans. Information Theory Vol. 22 (5) (1976) 644–654
5. Hietalahti, M.: Key Establishment in Ad-hoc Networks. Tik-110.501 Seminar on Network Security. (2000) 1–12
6. Hwang, M. S., Yang, W. P.: Conference key distribution schemes for secure digital mobile communications. IEEE J. Sel. Areas Comm. Vol. 13 (2) (1995) 416–420
7. Hwang, R. J., Lee, W. B., Chang, C. C.: A concept of designing cheater identification methods for secret sharing. J. Syst. Software Vol. 46 (1) (1999) 7–11
8. Ingemaresson, I., Tang, T. D., Wong, C.K.: A conference key distribution system. IEEE Trans. Information Theory. Vol. 28 (5) (1982) 714–720
9. Just, M., Vaudenay, S.: Authenticated multi-party key agreement. Advances in Cryptology Asiacrypt '96 Springer-Verlag, Berlin Heidelberg (1996) 36–49
10. Pieprzyk, J., Li, C. H.: Multi-party key agreement protocols. IEE Proc.-Comput. Digit. Tech. Vol. 147 No. 4 (2000) 229–236
11. Shamir, A.: How to share a secret. Communications of the ACM. Vol. 22 (1979) 612–613
12. Tseng, Y. M.: Multi-party Key agreement protocols with cheater identification. Applied Mathematics and Computation. Vol. 145 (2003) 551–559

Security of Shen et al.'s Timestamp-Based Password Authentication Scheme

Eun-Jun Yoon, Eun-Kyung Ryu, and Kee-Young Yoo

Department of Computer Engineering, Kyungpook National University,
Daegu 702-701, South Korea
{ejyoon, ekryu}@infosec.knu.ac.kr, yook@knu.ac.kr

Abstract. Recently, Shen et al. proposed an improvement on Yang-Shieh's timestamp-based password authentication scheme using smart cards. Then they claimed that their scheme cannot withstand a forged login attack, but also eliminate a problem of Yang-Shieh's. However, their scheme is still susceptible to forged login attack. In this paper, we show how the forged login attack can be worked out on Shen et al.'s scheme and present an enhancement to resolve such a problem.

Keywords: Cryptography, password, authentication, security, smart card.

1 Introduction

User authentication is an important part of security, along with confidentiality and integrity, for systems that allow remote access over untrustworthy networks, like the Internet. As such, a remote password authentication scheme authenticates the legitimacy of users over an insecure channel, where the password is often regarded as a secret shared between the remote system and the user. Based on knowledge of the password, the user can use it to create and send a valid login message to a remote system to gain the right to access. Meanwhile, the remote system also uses the shared password to check the validity of the login message and authenticate the user.

In 1999, Yang and Shieh [1] proposed a timestamp-based password authentication scheme using a smart card to achieve user authentication and arbitrarily change a password. In addition, the remote server does not need to store the passwords or verification tables for authentication the users. Subsequently, Chan and Cheng [2] pointed out that Yang and Shieh's scheme was vulnerable to a forged login attack, in which an intruder could impersonate legitimate users to login and accesses the remote server. In 2002, Fan et al. [3] also showed that Yang-Shieh scheme could not withstand the forged login attack and proposed a slight modification to eliminate the security flaws. Thereafter, in 2003, Shen et al. [5] pointed out that Fan et al.'s solution are inefficient and impractical because it limits the user identity ID_i with a strict form and proposed an improved scheme that could withstand the forged login attack and also provide mutual authentication to withstand the forged server attack. Yet, Shen et al.'s

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 665–671, 2004.
© Springer-Verlag Berlin Heidelberg 2004

improved scheme is still susceptible to forged login attack that developed by Sun et al. in [4] and Chen in [6], respectively. Accordingly, the current paper demonstrates that how the forged login attack can be worked out on their scheme, then present an enhancement to isolate such a problem.

The remainder of this paper is organized as follows: Section 2 briefly reviews Shen et al.'s scheme and demonstrates a forged login attack with their scheme. The proposed scheme is presented in Section 3, while Section 4 discusses the security of the proposed scheme. Some final conclusions are given in Section 5.

2 Review of the Shen et al.'s Timestamp-Based Password Authentication Scheme

This section briefly reviews Shen et al.'s scheme [5] and then show how the forged login attack can be worked out on their scheme.

2.1 Review of Shen et al.'s Scheme

Shen et al's timestamp-based password authentication scheme is composed of three phases; registration, login and authentication.

Registration phase: A new user U_i wants to register with a key information center (KIC) for accessing services. The KIC then performs the following steps:

1. User U_i securely submits his identity ID_i and a password PW_i to the KIC for registration.
2. Two large prime numbers p and q are generated, and let $n = p \cdot q$.
3. A prime number p is chosen at random as his public key, where p is relatively prime to $(p-1)(q-1)$.
4. An integer d as a corresponding secret key that satisfies $e \cdot d \equiv 1(\mathrm{mod}(p-1)(q-1))$.
5. An integer g, which is a primitive element in both d and $GF(q)$ is found, where g is KIC's public information.
6. Compute $S_i = ID_i^d \bmod n$ as U_i's secret information.
7. Compute h_i for U_i such that $h_i = g^{PW_i \cdot d} \bmod n$.
8. Compute $CID_i = f(ID_i \oplus d)$ where $\oplus$ stands for an exclusive operation.
9. Write $n, e, g, ID_i, CID_i, S_i, h_i, f(\cdot)$ into the smart card of U_i, and issue it through a secure channel.

Login phase: U_i must insert his smart card into the login device when he wants to login to the remote server. The smart card will perform the following operations after U_i keys in his identity ID_i and password PW_i.

1. Generate a random number r_i and compute X_i and Y_i as follows.

$$X_i = g^{r_i \cdot PW_i} \bmod n,$$
$$Y_i = S_i \cdot h_i^{r_i \cdot f(CID_i, T_C)} \bmod n.$$

Here T_C is the current date and time on the login device and $f(x, y)$ is a one-way function.
2. Send a message $M = \{ID_i, CID_i, X_i, Y_i, n, e, g, T_C\}$ to the remote server as a login request message.

Authentication phase: After receiving the login request message M from U_i, the remote server will perform the following operations to identify the login user:

1. Check the validity of ID_i. The remote server will reject the login request if the ID_i is incorrect.
2. Check the validity of T_C. If $(T_S - T_C) \geq \Delta T$, then the server rejects the login request. Here, T_S is the current date and time on the remote server; ΔT is the expected legitimate time interval for transmission delay.
3. Check the validity of CID_i by verifying $CID'_i = CID_i$, where $CID'_i = f(ID_i \oplus d)$.
4. Check the equation $(Y_i)^e = ID_i \cdot X_i^{f(CID'_i, T_C)} \bmod n$. If it holds, then the remote server accepts the user's login request and access.
5. Computes $R = (f(CID'_i, T_S))^d \bmod n$ for mutual authentication, where T'_S is the timestamp showing the current date and time from the remote server. It returns $M' = \{R, T'_S\}$ to the user U_i.

Upon receiving the message M' from the remote server, the user U_i verifies the message as follows:

6. Check the time interval between T'_S and T'_C, where T'_C is the date and time when the remote server receives the message M'. If $(T'_C - T'_S) \geq \Delta T$, then U_i rejects the remote server, where ΔT denotes the predetermined legitimate time interval of transmission delay.

7. Calculate R' using $R' = R^e \bmod n = (f(CID_i, T_S)^d)^e = f(CID_i, T_S)$. If $R' = f(CID_i, T_S)$ does not hold, U_i then rejects the remote server and breaks the connection.

2.2 Forged Login Attack on the Shen et al.'s Scheme

Unfortunately, Shen et al.'s scheme suffers from an authentication flaw similar to that developed by Sun et al. in [4] and Chen in [6], respectively. In Shen et al.'s scheme, any intruder can pretend to be a valid user U_i and login the remote server successfully by performing the following steps:

Step 1. Use the Extended Euclidean algorithm to compute $\gcd(e, f(CID_i, T_I)) = 1$, where T_I is the current timestamp. Note that the information $M = \{ID_i, CID_i, X_i, Y_i, n, e, g, T_C\}$ can be easily obtained by an intruder through wiretapping the communication channel between a legal user and the remote server.

Step 2. If $\gcd(e, f(CID_i, T_I)) = 1$, let a, b be the coefficients computed by the Extended Euclidean algorithm such that $a \cdot e + b \cdot f(CID_i, T_I)) = 1$. Because e is a prime number, so e and $f(CID_i, T_I)$ are relatively prime. Therefore, an intruder is easy to find a and b such $a \cdot e + b \cdot f(CID_i, T_I)) = 1$.
Step 3. Computes $X_f = (ID_i)^{-b} \bmod n$, $Y_f = (ID_i)^a \bmod n$.
Step 4. The forged login request message $M = \{ID_i, CID_i, X_f, Y_f, n, e, g, T_I\}$ is send to the remote server.

It is easy to verify that the above forged request message can pass the authentication in the Shen et al.'s scheme:

$$\begin{aligned}(Y_f)^e &= (ID_i)^{a \cdot e} (\bmod n) \\ &= (ID_i)^{1 - b \cdot f(CID_i, T_I)} (\bmod n) \\ &= ID_i \cdot ((ID_i)^{-b})^{f(CID_i, T_I)} (\bmod n) \\ &= ID_i \cdot (X_f)^{f(CID_i, T_I)} (\bmod n).\end{aligned}$$

This attack can be extended to consider $\gcd(e, f(CID_i, T_I)) = 2, 3, ...$instead of $\gcd(e, f(CID_i, T_I)) = 1$.

3 The Improved Scheme

This section proposes an improved password authentication scheme to overcome the above mentioned problem with Shen et al.'s scheme. Improved scheme is also composed of three phases; registration, login and authentication.

Registration phase: The proposed registration phase is the same as Shen et al's scheme and illustrated in Figure 1. A new user U_i wants to register with a key information center (KIC) for accessing services. The KIC then performs the following steps:

1. User U_i securely submits his identity ID_i and a password PW_i to the KIC for registration.
2. Two large prime numbers p and q are generated, and let $n = p \cdot q$.
3. A prime number p is chosen at random as his public key, where p is relatively prime to $(p-1)(q-1)$.
4. An integer d as a corresponding secret key that satisfies $e \cdot d \equiv 1 (\bmod (p-1)(q-1))$.
5. An integer g, which is a primitive element in both d and $GF(q)$ is found, where g is KIC's public information.
6. Compute $S_i = ID_i^d \bmod n$ as U_i's secret information.
7. Compute h_i for U_i such that $h_i = g^{PW_i \cdot d} \bmod n$.
8. Compute $CID_i = f(ID_i \oplus d)$ where $\oplus$ stands for an exclusive operation.
9. Write $n, e, g, ID_i, CID_i, S_i, h_i, f(\cdot)$ into the smart card of U_i, and issue it through a secure channel.

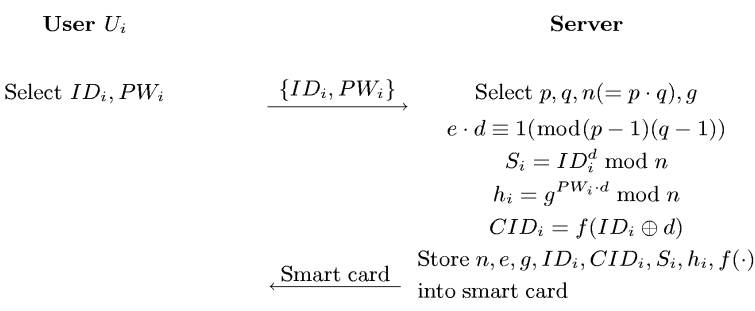


Fig. 1. The improved registration phase

Login phase: U_i must insert his smart card into the login device when he wants to login to the remote server. The smart card will perform the following operations after U_i keys in his identity ID_i and password PW_i.

1. Generate a random number r_i and compute X_i and U_i as follows.

$$\begin{aligned} X_i &= g^{r_i \cdot PW_i} \bmod n, \\ Y_i &= S_i \cdot h_i^{r_i \cdot f(CID_i, T_C)} \bmod n, \\ Z_i &= CID_i \oplus X_i, \\ V_i &= f(CID_i, X_i, Y_i). \end{aligned}$$

2. Send a message $M = \{ID_i, Y_i, Z_i, V_i, n, e, g, T_C\}$ to the remote server as a login request message.

Authentication phase: After receiving the login request message M from U_i, the remote server will perform the following operations to identify the login user:

1. Check the validity of ID_i. The remote server will reject the login request if the ID_i is incorrect.
2. Check the validity of T_C. If $(T_S - T_C) \geq \Delta T$, then the server rejects the login request.
3. Obtain X_i by computing $Z_i \oplus CID_i' = X_i$, where $CID_i' = f(ID_i \oplus d)$.
4. Check the validity of V_i by verifying $V_i = f(CID_i', X_i, Y_i)$. If it holds, then go to Step 5. Otherwise, login request is rejected.
5. Check the equation $(Y_i)^e = ID_i \cdot X_i^{f(CID_i', T_C)} \bmod n$. If it holds, then the remote server accepts the user's login request and access.
6. Computes $R = (f(CID_i', T_S'))^d \bmod n$ for mutual authentication, where T_S' is the timestamp showing the current date and time from the remote server. It returns $M' = \{R, T_S'\}$ to the user U_i.

Upon receiving the message M' from the remote server, the user U_i verifies the message as follows:

7. Check the time interval between T'_S and T'_C, where T'_C is the date and time when the remote server receives the message M'. If $(T'_C - T'_S) \geq \Delta T$, then U_i rejects the remote server, where ΔT denotes the predetermined legitimate time interval of transmission delay.
8. Calculate R' using $R' = R^e \bmod n = (f(CID'_i, T'_S)^d)^e = f(CID_i, T_S)$. If $R' = f(CID'_i, T'_S)$ does not hold, U_i then rejects the remote server and breaks the connection.

We show the modified login and authentication phase in Figure 2.

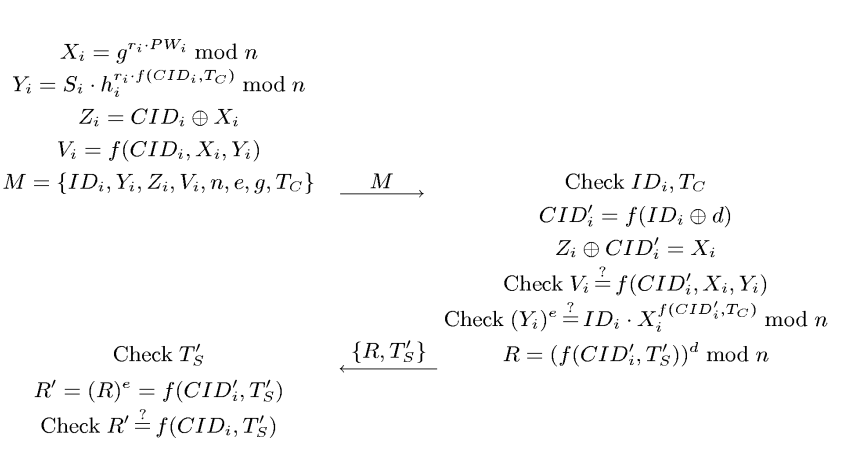


Fig. 2. The improved login and authentication phase

4 Security Analysis

The following analyzes the security of the proposed scheme:

Forged login attack: The forged login attack could succeed in the Shen et al.'s scheme because the intruder can generate a legitimate X_i freely. But our proposed scheme can withstand the forged login attack. Let us consider the following scenario, the intruder can intercept $M = \{ID_i, Y_i, Z_i, V_i, n, e, g, T_C\}$ sent by the user U_i in Step 1, then uses it to impersonate the user when sending the next login message. However, such a modification will fail in Steps 4, 5 of the authentication phase, because an intruder has no way of obtaining the values CID_i, X_i to compute the valid parameter Z_i.

Replay attack: For replay attacks, neither the replay of an old message $M = \{ID_i, Y_i, Z_i, V_i, n, e, g, T_C\}$ in the login phase of the authentication phase will work, as it will fail in Steps 4, 5, and 6 of the authentication phase due to the time interval $(T_S - T_C) \geq \Delta T$.

Secret key guessing attack: Given a valid request message $M = \{ID_i, Y_i, Z_i, V_i, n, e, g, T_C\}$, it is infeasible that an intruder can compute d using equation $Y_i = S_i \cdot h_i^{r_i \cdot f(CID_i, T_C)} \bmod n$, because it is a one-way property of a secure one-way function and a discrete logarithm problem. Even if the smart card of the user U_i is picked up by the intruder, it is still difficult for the intruder to derive d.

Forged server attack: Our proposed scheme can withstand the forged server attack because added mutual authentication. If a masquerader server tries to cheat the requesting user U_i, it has to prepare a valid message $M' = \{R, T'_S\}$. However, this is infeasible, there in no way to derive the value CID'_i to compute the value $R = (f(CID'_i, T'_S))^d \bmod n$ without the knowledge of server's secret key d. In addition, a replay message can be exposed because of the time stamp.

5 Conclusion

In this paper, we have demonstrated that Shen et al.'s timestamp-based password authentication scheme is vulnerable to a forged login attack. We have also presented an enhancement to solve such a problem, by providing a mutual authentication between the user and a remote system. The proposed scheme is practical and efficient timestamp-based password authentication scheme than other schemes [1][3][5].

Acknowledgements. We would like to thank the anonymous reviewers for their helpful comments. This work was supported by the Brain Korea 21 Project in 2003.

References

1. W. H. Yang, and S. P. Shieh, "Password authentication scheme with smart cards," Computers & Security, vol. 18, no. 8, pp. 727-733, 1999.
2. C. K. Chan and L. M. Cheng, "Cryptanalysis of timestamp-based password authentication scheme," Computers & Security, vol. 21, no. 1, pp. 74-76, 2002.
3. L. Fan, J. H. Li, and H. W. Zhu, "An enhancement of timestamp-based password authentication scheme," Computers & Security, vol. 21, no. 7, pp. 665-667, 2002.
4. H. M. Sun and H. T. Yeh, "Further cryptanalysis of a password authentication scheme with smart cards," IEICE Transactions on Communications, vol. E86-B, no. 4, pp. 1412-1415, 2003.
5. J. J. Shen, C. W. Lin and M. S. Hwang, "Security enhancement for the timestamp-based password authentication scheme using smart cards," Computers & Security, vol. 22, no. 7, pp. 591-595, 2003.
6. K. F. Chen, "Attacks on the (enhanced) Yang-Shieh authentication," Computers & Security, vol. 22, no. 8, pp. 725-727, 2003.

ID-Based Authenticated Multiple-Key Agreement Protocol from Pairings

Kee-Won Kim, Eun-Kyung Ryu, and Kee-Young Yoo

Department of Computer Engineering, Kyungpook National University, Daegu, KOREA, 702-701
{nirvana,ekryu}@infosec.knu.ac.kr, yook@knu.ac.kr

Abstract. To achieve secure data communications, participants should be authenticated and a new session key must be agreed securely. An authenticated key agreement protocol combining the meaning of user authentication and key agreement is necessary for these purposes. This paper proposes a new ID-based multiple-key agreement protocol. The authenticity of the protocol is provided by a signature scheme. The proposed protocol allows two parties to establish n^2 common secret keys if they compute and send n Diffie-Helman's public keys. The security attributes of the proposed protocol are examined using heuristic methods.

1 Introduction

Key agreement is one of the fundamental cryptographic primitives after encryption and digital signatures. Diffie-Hellman key agreement protocol [1] is the well-known method to enable two parties to establish a secret session key by exchanging messages over an insecure channel. Its security is based on the intractablility of the Diffie-Hellman problem and the discrete logarithm problem. However, the main problem of the Diffie-Hellman key agreement protocol is not resistant to the man-in-the-middle attack since it does not provide user authentication.

In 1995, Menezes et al. [2] proposed the MQV key agreement protocol, which is the first key agreement protocol that used a signature for the Diffie-Hellman public keys without using a one-way hash function. The IEEE P1363 committee has adopted the MQV key agreement protocol to become a standard [3]. Based on the MQV protocol, in 1998 Harn and Lin [4] proposed a authenticated multiple-key agreement protocol without using one-way hash function to enable two communication parties to establish multiple common secret keys in a single round of message exchange. Plenty of multiple-key agreement protocol [4,5,6,7, 8,9] have been presented after the Harn and Lin's works. Most of them adopt certificate to provide the authentication of the long-term public key.

However, in a certificate system, the participants must first verify the certificate of the user before using the public key of an user. As a consequence, this system requires a large amount of computing time and storage. In 1984, Shamir [10] asked for identity-based encryption and signature schemes to simplify key management procedures in certificate-based public key infrastructure.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 672–680, 2004.
© Springer-Verlag Berlin Heidelberg 2004

Since then, many ID-based encryption schemes and signature schemes have been proposed [11,12,13]. The idea of ID-based cryptosystems is that the identity information of an user functions as his public key. A key generation center which is trusted by all users is responsible for the generation of users' corresponding private keys. The bilinear pairings, namely the Weil pairing and the Tate pairing of algebraic curves, are important tools for construction of ID-based cryptographic schemes. Many ID-based cryptographic schemes [11,12,13,14,15,16,17,18,19,20] have been proposed using the bilinear pairings.

Recently, there have been proposed several ID-based key agreement protocols [14,18,19]. Also several ID-based tripartite authenticated key agreement protocols were proposed [15,16,17,20]. In this paper, we propose a new ID-based authenticated multiple-key agreement protocol. The security analysis is presented. Also, we provide its security analysis. The remainder of the paper is organized as follows. Section 2 briefly discusses the properties of secure key agreement protocols. Section 3 discusses the mathematical definitions and preliminaries required for the new protocol. The new protocol for ID-based authenticated multiple-key agreement is proposed and the security analysis is presented in Section 4. Section 5 concludes the paper.

2 Properties of Key Agreement Protocols

A secure protocol should be able to withstand both passive attacks and active attacks. A number of desirable attributes of key agreement protocols have been identified(See [21,22] for further details).

1. **Know-key Security**. A protocol is known-key secure if it still achieves its goal in the face of an adversary who has learned some previous session keys.
2. **(Perfect) Forward Secrecy.** If long-term private keys of one or more entities are compromised, the secrecy of previous session keys established by honest entities is not affected.
3. **Key-compromise Impersonation Resilience.** Suppose A's long-term private key is disclosed. Then of course an adversary can impersonate A in any protocol in which A is identified by this key. We say that a protocol resists key-compromise impersonation when this loss does not enable an adversary to impersonate other entities as well.
4. **No Unknown Key-share.** Entity A cannot be coerced into sharing a key with entity B without A's knowledge, i.e., when A believes the key is shared with some entity $C \neq B$, and B (correctly) believes the key is shared with A.
5. **No Key Control.** It should be not be possible for any of the participants (or an adversary) to force the session key to a preselected value or predict the value of the session key.

Desirable performance attributes of key agreement protocols include a minimal number of rounds, low communication overhead, and low computation overhead.

3 Preliminaries

In this section, we give the basic concept and some properties of the bilinear pairings. This section also present ID-based public key infrastructure based on pairing.

3.1 Basic Concepts on Bilinear Pairings

Let G_1 be a cyclic additive group generated by P, whose order is a prime q, and G_2 be a cyclic multiplicative group of the same order q. We assume that the discrete logarithm problems (DLP) in both G_1 and G_2 are hard. Let $e : G_1 \times G_1 \rightarrow G_2$ be a pairing which satisfies the following conditions (examples are the Weil and Tate pairings associated with supersingular elliptic curves):

1. Bilinear

$$e(P_1 + P_2, Q) = e(P_1, Q)e(P_2, Q)$$
$$e(P, Q_1 + Q_2) = e(P, Q_1)e(P, Q_2)$$
$$i.e., e(aP, bQ) = e(P, Q)^{ab} \text{ where } a, b \in Z_q^*, P, Q \in G_1.$$

2. Non-Degenerate
   There exists $P \in G_1$ and $Q \in G_1$ such that $e(P, Q) \neq 1$.
3. Computable:
   There is an efficient algorithm to compute $e(P, Q)$ for all $P, Q \in G_1$.

A more comprehensive description is provided in [12]. Now we describe two mathematical problems.

Computational Diffie-Hellman Problem(CDHP): Let G_1 be a cyclic additive group generated by P, whose order is a prime q. The CDHP in G_1 is as follows:

For $a, b \in Z_q^*$, given P, aP, bP, compute abP.

Bilinear Diffie-Hellman Problem(BDHP): Let G_1, G_2 be two groups of prime order q. Let $e : G_1 \times G_1 \rightarrow G_2$ be a bilinear map and let P be a generator of G_1. The BDHP in (G_1, G_2, e) is as follows:

For $a, b, c \in Z_p^*$, given (P, aP, bP, cP), compute $W = e(P, P)^{abc} \in G_2$.

We assume through this paper that BDHP is hard, which means there is no polynomial time algorithm to solve BDHP with non-negligible probability.

3.2 ID-Based Public Key Infrastructure

The ID-based public key infrastructure, introduced by Shamir [10], allows some public information of the user such as name, address and email etc., rather than arbitrary string to be used his public key. The private key of the user is calculated

by KGC and sent to the user via a secure channel. The basic operations consist of set up and private key extraction. When we use bilinear pairings to construct ID-based public key infrastructure, set up and private key extraction can be implemented as follows:

Let G_1 be a cyclic additive group generated by P, whose order is a prime q, and G_2 be a cyclic multiplicative group of the same order q. A bilinear pairings is a map $e : G_1 \times G_1 \to G_2$. Define two cryptographic hash functions $H_1 : \{0,1\}^* \to G_1$ and $H_2 : G_1 \to Z_q^*$.

- **Setup :** KGC chooses a random number $s \in Z_q^*$ and set $P_{KGC} = sP$. KGC publishes system parameters *paramas*$= \{q, G_1, G_2, e, P, P_{KGC}, H_1, H_2\}$, and keeps s as the *master-key*, which is known only by itself.
- **Private Key Extraction :** An user submits his identity information ID to KGC. KGC computes the user's public key as $Q_{User} = H_1(ID_{User})$, and returns $S_{User} = sQ_{User}$ to the user as his private key.

4 ID-Based Authenticated Multiple-Key Agreement Protocol

In this section, we propose a new authenticated multiple-key agreement protocol, and then examine the security.

4.1 Protocol

Suppose two users A and B wish to agree multiple common session keys. The identities of A and B are ID_A and ID_B. With the ID-based public key infrastructure, the long-term public key and the long-term private key for A are $Q_A = H_1(ID_A)$ and $S_A = sQ_A$, respectively. Similarly, B has $Q_B = H_1(ID_B)$ and $S_B = sQ_B$. The long-term private keys of users have been obtained from the key generation center. The public key of key generation center is $P_{KGC} = sP$. For simplicity, let us assume that A and B want to share four common session keys. The protocol is as follows.

Step 1:A selects two random integers a_1 and a_2, called the short-term private keys. Furthermore, A computes two short-term public keys $T_{A1} = a_1P$ and $T_{A2} = a_2P$. A obtains the signature value V_A by computing the following equation:

$$V_A = H_2(T_{A1})H_2(T_{A2})S_A + (a_1 + a_2)P_{KGC}. \tag{1}$$

Finally, A sends the authenticated messages $\{T_{A1}, T_{A2}, V_A\}$ to B.

Step 2: B selects two random integers b_1 and b_2, called the short-term private keys. Furthermore, B computes two short-term public keys $T_{B1} = b_1P$ and $T_{B2} = b_2P$. B obtains the signature value V_B by computing the following equation:

$$V_B = H_2(T_{B1})H_2(T_{B2})S_B + (b_1 + b_2)P_{KGC}. \tag{2}$$

Finally, B sends the authenticated messages $\{T_{B1}, T_{B2}, V_B\}$ to A.

Step 3: A verifies the message $\{T_{B1}, T_{B2}, V_B\}$ sent from B by checking the following verification equation:

$$e(V_B, P) \stackrel{?}{=} e(H_2(T_{B1})H_2(T_{B2})Q_B + T_{B1} + T_{B2}, P_{KGC}). \tag{3}$$

If the verification is valid, A uses T_{B1} and T_{B2} to compute four common session keys as follows:

$$\begin{aligned} K_1 &= e(P_{KGC}, T_{B1})^{a_1} = e(P, P)^{a_1 b_1 s} \\ K_2 &= e(P_{KGC}, T_{B2})^{a_1} = e(P, P)^{a_1 b_2 s} \\ K_3 &= e(P_{KGC}, T_{B1})^{a_2} = e(P, P)^{a_2 b_1 s} \\ K_4 &= e(P_{KGC}, T_{B2})^{a_2} = e(P, P)^{a_2 b_2 s}. \end{aligned} \tag{4}$$

Step 4: B verifies the message and signature similarly by checking the verification equation:

$$e(V_A, P) \stackrel{?}{=} e(H_2(T_{A1})H_2(T_{A2})Q_A + T_{A1} + T_{A2}, P_{KGC}). \tag{5}$$

Finally, B also computes four common session keys as follows:

$$\begin{aligned} K_1 &= e(P_{KGC}, T_{A1})^{b_1} = e(P, P)^{a_1 b_1 s} \\ K_2 &= e(P_{KGC}, T_{A1})^{b_2} = e(P, P)^{a_1 b_2 s} \\ K_3 &= e(P_{KGC}, T_{A2})^{b_1} = e(P, P)^{a_2 b_1 s} \\ K_4 &= e(P_{KGC}, T_{A2})^{b_2} = e(P, P)^{a_2 b_2 s}. \end{aligned} \tag{6}$$

The following shows that the above protocol works correct.

Theorem 1. *If the equation* $e(V_A, P) \stackrel{?}{=} e(H_2(T_{A1})H_2(T_{A2})Q_A + T_{A1} + T_{A2}, P_{KGC})$ *holds, B may verify the message* $\{T_{A1}, T_{A2}, V_A\}$ *sent from A.*

Proof. Since $T_{A1} = a_1P$, $T_{A2} = a_2P$ and $V_A = H_2(T_{A1})H_2(T_{A2})S_A + (a_1 + a_2)P_{KGC}$, we have

$$\begin{aligned} e(V_A, P) &= e(H_2(T_{A1})H_2(T_{A2})S_A + (a_1 + a_2)P_{KGC}, P) \\ &= e(s(H_2(T_{A1})H_2(T_{A2})Q_A + (a_1 + a_2)P), P) \\ &= e(H_2(T_{A1})H_2(T_{A2})Q_A + T_{A1} + T_{A1}, P_{KGC}). \end{aligned} \tag{7}$$

Thus, the equation $e(V_A, P) = e(H_2(T_{A1})H_2(T_{A2})Q_A + T_{A1} + T_{A1}, P_{KGC})$.

4.2 Security of the Protocol

The principle of the proposed protocol is that the short-term private keys a_1, a_2, $b1$ and b_2 of the two entities determines the four session keys $K_1 = e(P,P)^{a_1 b_1 s}$, $K_2 = e(P,P)^{a_1 b_2 s}$, $K_3 = e(P,P)^{a_2 b_1 s}$ and $K_4 = e(P,P)^{a_2 b_2 s}$. Due to the bilinearity of the pairing, the two entities don't have to exchange short-term private keys over secure channels. They exchange $\{T_{A1},\ T_{A2}\}$ and $\{T_{B1},\ T_{B2}\}$ publicly to determine the session keys instead. The secrecy of the session keys relies on the assumption of hardness of BDHP, *i.e.*, given P, aP, bP and sP, it is hard to determine $e(P,P)^{abs}$.

The authenticity of $\{T_{A1}, T_{A2}\}$ and $\{T_{B1}, T_{B2}\}$ is achieved by attaching signatures V_A and V_B, respectively. The signatures are computed by the entities using their long-term private key. As a consequence, the authenticity of the multiple key agreement protocol is assured by the security of the following signature scheme, which relies on the ID-based public key infrastructure introduced in Section 2. Without loss of generality, we take entity A as the signing entity. Signer A has a long-term private key S_A, while the long-term public key is Q_A.

Signing: Suppose that the messages to be sent is $m = NULL$. Signer A randomly chooses two integers $a_1, a_2 \in Z_q^*$. He computes $T_{A1} = a_1 P$, $T_{A1} = a_1 P$ and $V_A = H_2(T_{A1})H_2(T_{A2})S_A + (a_1 + a_2)P_{KGC}$. Then the signature of m is (T_{A1}, T_{A2}, V_A).

Verification: After getting a message m and its signature (T_{A1}, T_{A2}, V_A), the verifier accepts the signature if and only if the following equation holds

$$e(V_A, P) \stackrel{?}{=} e(H_2(T_{A1})H_2(T_{A2})Q_A + T_{A1} + T_{A2}, P_{KGC}). \tag{8}$$

Our signature scheme is secure against existential forgery under an adaptively chosen message attack in the random oracle model. The proof is similar to that of scheme 3 in [23].

Now we give a brief security analysis to show that the above signature scheme is secure against existential forgery. Suppose that there is a polynomial time probabilistic Turing machine E which takes m and Q_A as input, and output an existential forgery of a signature from A with a non-negligible probability.

By the Forking Lemma of Pointcheval and Stern [24], E may get two forgeries of signature from A for the same message m within a polynomial time. Let the two signature forgeries for m be (T_{A1}, T_{A2}, V_A) and (T_{A1}, T_{A2}, V'_A). We have

$$V_A = H_2(m, T_{A1})H_2(m, T_{A2})S_A + (a_1 + a_2)P_{KGC} \tag{9}$$

and

$$V'_A = H'_2(m, T_{A1})H'_2(m, T_{A2})S_A + (a_1 + a_2)P_{KGC}. \tag{10}$$

It follows that

$$V_A - V'_A = (H_2(m, T_{A1})H_2(m, T_{A2}) - H'_2(m, T_{A1})H'_2(m, T_{A2}))S_A \tag{11}$$

hence

$$e(V_A - V'_A, P) = e(Q_A, P_{KGC})^{H_2(m,T_{A1})H_2(m,T_{A2}) - H'_2(m,T_{A1})H'_2(m,T_{A2})}. \quad (12)$$

The above equation means that for a random element $v \in G_2$, an element $R \in G_1$ can be found such that $e(R, P) = v$ in a polynomial time. There is a polynomial time algorithm $i : G_2 \to G_1$ inverting the pairing e, that is $x = e(i(x), P)$. Let g be a generator of G_2. Then $t = e(i(g), i(g))$ is also a generator of G_2. Furthermore, $e(i(g^l), i(g^m)) = t^{lm}$. That is, given g^l and g^m we have computed t^{lm} and have hence solved an instance of the weak Diffie-Hellman problem in G_2. By [25] we can now conclude that inverting the pairing $e(R, P)$ is at least as hard as solving the Diffie-Hellman problem

A secure key agreement protocol should be able to withstand the passive and active attacks and has a number of desirable security attributes described in Section 2.

Known-key Security: In view of the randomness of a_1, a_2, b_1 and b_2 in our protocol, session keys in different key agreements are independent of each other. The knowledge of previous session keys does not help an adversary to derive any future session key. Hence, our protocol has the property of known-key security.

Perfect Forward Secrecy: Suppose that long-term private keys S_A and S_B are compromised. The adversary who knows these values cannot compute $e(P, P)^{a_1 b_1 s}$, $e(P, P)^{a_1 b_2 s}$, $e(P, P)^{a_2 b_1 s}$ and $e(P, P)^{a_2 b_2 s}$, since she still faces the Diffie-Hellman problem to calculate them: compute $e(P, P)^{a_1 b_1 s}$ from $e(T_{A1}, T_{B1}) = e(P, P)^{a_1 b_1}$ and $e(P_{KGC}, P) = e(P, P)^s$ (or $e(T_{A1}, P_{KGC}) = e(P, P)^{a_1 s}$ and $e(T_{B1}, P) = e(P, P)^{b_1}$). To learn the previous session keys, the adversary has to get the corresponding short-term private keys. Note that compromise of the key generation center's master-key s will allow anyone to compute the key via $e(T_{A1}, T_{B1})^s$, $e(T_{A1}, T_{B2})^s$, $e(T_{A2}, T_{B1})^s$ and $e(T_{A2}, T_{B2})^s$. Thus the key generation center is able to recover the agreed session keys from the message flows and its secret key. Combined with a secret sharing scheme for the key generation center's secret key, this allows for an efficient ID-based key escrow facility for sessions.

Key-compromise Impersonation Resilience: Suppose A's long-term private key S_A is disclosed. Suppose that an adversary who knows the value wants to masquerade B to A. First, she chooses two short-term private keys b_1 and b_2 and computes $T_{B1} = b_1 P$ and $T_{B2} = b_2 P$. But she cannot compute signature $V_B = H_2(T_{B1})H_2(T_{B2})S_B + (b_1 + b_2)P_{KGC}$ since she does not know S_B. Therefore, our protocol is resistant to the key-compromise impersonation attack.

No Unknown Key-share: To implement unknown key-share attack on our protocol, the adversary is required to learn the short-term private key of some entity. Otherwise, the attack hardly works. Hence, we claim that our protocol has the attribute of no unknown key-share.

No Key Control: Neither entity should be able to force the session key to a preselected value. In our protocol, $\{T_{A1}, T_{A2}\}$ and $\{T_{B1}, T_{B2}\}$ are generated by two users, and thereby none of them can predetermine the session key or force the session key to a preselected value. Therefore, our protocol has the property of key control.

5 Conclusions

ID-based public key infrastructure can be an alternative for certificate-based public key infrastructure, especially when efficient key management and moderate security are required. In this paper, we proposed an ID-based authenticated multiple-key agreement protocol. The resulting key is determined by the short-term keys of the two entities. The proposed protocol allows two parties to establish n^2 common secret keys if they compute and send n Diffie-Helman's public keys. The authenticity of the protocol is assured by a digital signature scheme. The signature scheme resists existential forgeries against adaptively chosen message attacks. Our protocol has the following security attributes: known session key security, perfect forward secrecy, no key-compromise impersonation, no unknown key-share, and no key control.

Acknowledgment. This work was supported by the Brain Korea 21 Project in 2003.

References

1. Diffie, W. and Hellman, M.E.: New Directions in Cryptography. IEEE Transactions on Information Theory, IT-22, (1976) 644–654
2. Menezes, A.J., Qu, M. and Vanstone, S.A.: Some Key Agreement Protocols Providing Implicit Authentication, 2nd Workshop Selected Area in Cryptography, SAC '95, (1995) 22–32
3. IEEE P1363 Working Group, 2001, IEEE P1363a D10 (Draft version 10): Standard Specifications for Public Key Cryptography: Additional Techniques, IEEE P1363 Working Group, Working draft (available from http://grouper.ieee.org/groups/1363)
4. Harn, L. and Lin, H.Y.: An Authenticated Key Agreement Protocol without Using One-way Function. Proceedings of 8th National Conference Information Security. (1998) 155–160
5. Harn, L. and Lin, H.-Y.: Authenticated Key Agreement without Using One-way Hash Functions. Electronics Letters, Vol. 37, No. 10, (2001) 229–630
6. Tseng, Y.-M: Robust Generalized MQV Key Agreement Protocol without Using One-way Hash Functions. Computer Standards and Interfaces, Vol. 24, (2002) 241–246
7. Shao, Z.: Security of Robust Generalized MQV Key Agreement Protocol Without Using One-way Hash Functions, Computer Standards and Interfaces, Vol. 25, (2003) 431–436

8. Hwan, R.-J, Shiau, S.-H. and Lai, C.-H.: An Enhanced Authentication Key Exchange Protocol, Proceedings of the 17th International Conference on Advanced Information Networking and Applications, AINA 2003, (2003) 202–205
9. Chien, H.-Y. and Jan, J.-K: Improved Authenticated Multiple-key Agreement Protocol Without Using Conventional One-way Function, Applied Mathematics and Computation, Vol. 147 (2004) 491–497
10. Shamir, A.: Identity-based Cryptosystems and Signature Schemes. Advances in Cryptology, Crypto 84, Lecture Notes in Computer Sciene, Vol. 196. Springer-Verlag. (1984) 47–53
11. Tsuji, S. and Itoh, T.: An ID-based Cryptosystem Based on the Discret Logarithm Problem, IEEE Journal of Selected Areas in Communications, Vol. 7, No.4, (1989) 467–473
12. Boneh, D. and Franklin, M.: Identity-based Encryption from the Weil Pairing. Advances in Cryptology, Crypto 2001, Lecture Notes in Computer Sciene, Vol. 2139. Springer-Verlag. (2001) 213–229
13. Cocks, C.: An Identity Based Encryption Scheme Based on Quadratic Residus. Crytography and Coding, Lecture Notes in Computer Sciene, Vol. 2260. Springer-Verlag. (2001) 360–363
14. Smart, N.P.: Identity-based Authnticated Key Agreement Protocol Based on Weil Pairing, Electronics Letters, Vol. 38, No. 13, (2002) 630–632
15. Al-Riyami, S. and Paterson K.G.: Authenticated Three Party Key Agreement Protocols from Pairings. Cryptology ePrint Archive, Report 2002/035, available at http://eprint.iacr.org/2002/035/ (2002)
16. Zhang, F., Liu, S. and Kim, K.J.: ID-based One Round Authenticated Tripartite Key Agreement Protocol with Pairings. Cryptology ePrint Archive, Report 2002/122, available at http://eprint.iacr.org/2002/122/ (2002)
17. Nalla, D. and Reddy, K.C.: ID-based Tripartite Authenticated Key Agreement Protocols from Pairings. Cryptology ePrint Archive, Report 2003/004, available at http://eprint.iacr.org/2003/004/ (2003)
18. Shim, K.: Efficient ID-based Authenticated Key Agreement Protocol Based on Weil Pairing. Electronics Letters, Vol. 39, No. 8, (2003) 653–654
19. Yi, X.: Efficient ID-based Key Agreement from Weil Pairing. Electronics Letters, Vol. 39, No. 2, (2003) 206–208
20. Nalla, D.: ID-based Tripartite Key Agreement with Signatures. Cryptology ePrint Archive, Report 2003/144, available at http://eprint.iacr.org/2003/144/ (2003)
21. Blake-Wilson, S., Johnson, D. and Menezes, A.: Key Agreement Protocols and Their Security Analysis. Proceedings of the sixth IMA International Conference on Cryptography and Coding, Lecture Notes in Computer Sciene, Vol. 1355. Springer-Verlag. (1997) 310–324
22. Law, L., Menezes, A.J., Qu, M., Solinas, J., and Vanstone, S.: An Efficient Protocol for Authenticated Key Agreement. Designs, Codes and Cryptography, Vol. 28. (2003) 119–134
23. Hess, F.: Efficient Identity based Signature Schemes Based on Pairings. Proceedings of 9th Workshop on Selected Areas in Cryptography, SAC 2002, Lecture Notes in Computer Sciene, Vol. 2595. Springer-Verlag. (2003) 310–324
24. Pointcheval, D. and Stern, J.: Security Arguments for Digital Signatures and Blind Signatures, Journal of Cryptology, No. 13, (2000) 361–396
25. Verheul, E.R.: Evidence that XTR is more secure than supersingular elliptic curve cryptosystems. Advances in Cryptology, EuroCrypt 2001, Lecture Notes in Computer Sciene, Vol. 2045. Springer-Verlag. (2001) 195–210

A Fine-Grained Taxonomy of Security Vulnerability in Active Network Environments*

Jin S. Yang[1], Young J. Han[1], Dong S. Kim[1], Beom H. Chang[2], Tai M. Chung[1], and Jung C. Na[2]

[1] Internet Management Technology Laboratory and
Cemi: Center for Emergency Medical Informatics,
Scool of Information and Communication Engineering, SungKyunKwan University,
300 Cheoncheon-dong, Jangan-gu, Suwon-si, Gyeonggi-do, 440-746, Korea
{jsyang, yjhan, dskim, tmchung}@imtl.skku.ac.kr
[2] Network Security Dept., Information Security Research Div.,
Electronics and Telecommunications Research Institute, Korea
{bchang, njc}@etri.re.kr

Abstract. Active networks support infrastructure that the routers or switches of the network perform customized computations on the messages flowing through them. For the active networking, it is necessary to build the new components: NodeOS (Node Operation System), Execution Environment (EE), Active Application (AA). The addition of the new components occurs potentially security vulnerability. Although studies have been made existing components from external threat in active network environments, taxonomy of security vulnerability of active network components has never been studied so far. Therefore, there is no criterion for these vulnerabilities in active network environments. In this paper, we analyze active network components for vulnerability scanning and classify vulnerabilities based on active network components. This taxonomy presents the criterion of security vulnerabilities in active network environments.

1 Introduction

Active networks technologies present a new direction for more flexible and faster service deployment. Active networks support infrastructure that the routers or switches of the network perform customized computations on the messages flowing through them. In order to execute programs in intermediate node, it is essential to embed active network components.

Active networks are based on existent technologies. Therefore, by adding new components and existent technologies, security vulnerabilities can be accomplished. These vulnerabilities should be seriously considered in such a side that shared devices as router permits user defined processing. The research about

* This study was partially supported by a grant of the Korea Health 21 R&D Project, Ministry of Health & Welfare, Republic of Korea(02-PJ3-PG6-EV08-0001)

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 681–688, 2004.
© Springer-Verlag Berlin Heidelberg 2004

active network security goes on protecting active network components from external threats. There is no research about vulnerability of active network components. Therefore, by adding new components, security vulnerabilities in active network environments must be analyzed and classified differently.

To secure a system from external threats, it is necessary to understand and analysis what the features exist in vulnerable system. The Section 2 describes active networks and existing scheme of fault classification as related work. The Section 3 describes motivation of taxonomy of security vulnerability in active network environments. The Section 4 explains security elements and their considerations in active network components before vulnerability classification. The Section 5 describes the vulnerability classification in that consideration of security elements is applied.

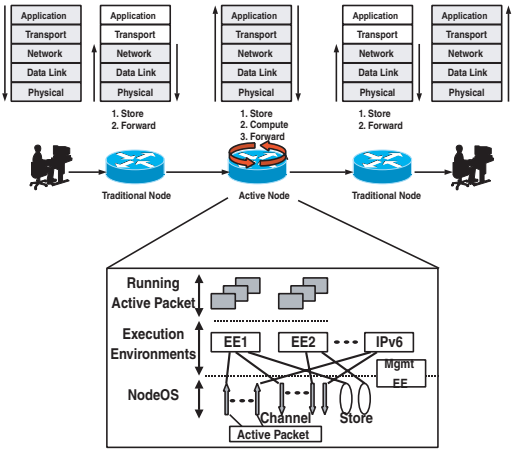


Fig. 1. The concept of active networks

2 Related Works

2.1 Active Networks

The concept of active networking has emerged from discussions within the DARPA (Defense Advanced Research Projects Agency) research community in 1994 and 1995 on the future directions of networking systems. Active networks are a novel approach to network architecture in which the switches of the network perform customized computations on the messages flowing through them. Routers can perform computations on user data and while packets can carry programs to be executed on routers and possibly change their state. It supports flexibility, efficiency, and fast deployment of new services [4], [6], [7], [10]. Figure 1 shows the concept of active networks. Active Node (AN) consists of NodeOS, EEs, and AAs. Role of each components are as follows [2], [11].

NodeOS. NodeOS multiplexes the node's communication, memory and computational resources among the various packet flows that traverse the node. In order to multiplex, NodeOS defines five primary abstractions: Domain, Thread Pool, Memory Pool, Channels, and Files. As mentioned above, NodeOS acts important roles, such as process management, memory management, channel management, and so forth [2].

EE. One or more EEs define a particular programming model for writing AA. EE with concept as Java Virtual Machine, supports limited programming environments for execution of AA. EE supports access control of main resources for AA, setting a security policy about node, setting alteration about existent EE, and so forth [5], [12].

AA. AA is a user defined service and can be implemented by a program that has various kinds of function according to purpose of service, and can be implemented in the various kinds of programming languages according to EE.

2.2 Existing Schemes of Fault Classification

The Protection Analysis (PA) project researches on protection errors in operating systems (OS). The PA project proposed four representative categories of faults. These were designed to group faults based on their syntactic structure and are too broad to be used for effective data organization. The RISOS project was a study of computer security and privacy. The fault categories proposed in the RISOS project are general enough to classify faults from several OS, but the generality of the fault categories prevents fine-grain classification and can lead to ambiguities, classifying the same fault in more than one category. Carl Landwehr and Bran marick published a collection of security faults. Although the results of the studies are insightful, the classification scheme provided is not suitable for data organization and fault categories are ambiguous [15]. As mentioned above, classification of vulnerability must have criterion that should be clear. Fault categories are not too broad and general. Also, classification of vulnerability must use easily.

3 Motivations

Vulnerability classification is required to risk management. And classified vulnerability can be used usefully in vulnerability scanning. Active network components are elements added in existent infrastructure. They can cause new vulnerabilities potentially. Therefore, these vulnerabilities and traditional vulnerabilities should be classified differently. The criterion of this paper from vulnerability classification is based on the feature of active network components. The feature of active network components based on taxonomy is very important. In tradition network environments, when vulnerabilities were found on components-OS, software, and so forth-developers or development institutions for the components are responsibility for reconfiguration or patch of components about vulnerabilities. The

points that describe above are more important in active network environments. Because active networks permits processing in intermediate node. Vulnerabilities can cause huge affect to whole network as well as relevant node.

4 Security Components of Active Network Environments

Traditional OS used in AN [8]. For examples, ANTS project uses Linux, Smartpacket project uses freeBSD, FAIN project uses solaris, and so forth.

Traditional OS includes resource management mechanisms and security mechanisms [1]. NodeOS, like traditional OS, includes them. But, the management objects of NodeOS are active network components. There are vulnerabilities that exist with traditional OS used at AN [16], [17]. AN that use traditional OS includes all of them. In this paper, the vulnerabilities of traditional OS is excepted in classifying objects but is used in vulnerability analysis. Because [3] describes "Attacks in an active network will arise from the same actions as in traditional networks". Security consideration of each components are as follows.

NodeOS. Trivial security vulnerabilities of AN can cause critical damages to other components and network services in AN. Therefore, we must analyze the threats of core components, that is primary abstractions [2].

EE. Running EE, exception situations that can appear are exceptions of code transmission, of unauthenticated and malicious code, and of compatibility between EEs. Taxonomy of security vulnerability must consider these exceptions running on EEs in AN. EEs must consider vulnerability of programming language, too.

AA. Security in AA must be considered authentication of application by itself, side effect by authentication failure, rate limitation and language vulnerability. Figure 2 shows threat correlation between active network components [3].

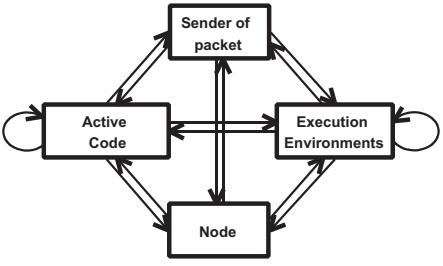


Fig. 2. Threat correlation between active network components

Key points in Figure 2 are EE and AC. Because each component will be installed dynamically, EE and AC can threat by themselves.

5 Taxonomy of Security Vulnerability

5.1 Terminologies

Before classifying security vulnerabilities in active network environments, We must define some terminologies. Existent vulnerability classification used fault or error Usually. But We will use only exception. Meaning of terminologies is as follows.

Fault: System problem has indwelled
Error: System problem is detected to face
Exception: Unexpected situation about any problem

It is because can't speak Fault or Error definitely without testing. When we classify security vulnerability, we use Exception in this paper. Exception in this paper means "*presumable*" or "*possible*".

5.2 Taxonomy

This paper describes vulnerability in only active network infrastructure. We classify NodeOS, EE, and AP based on security components of active networks. Because it is based on intuitional classification. These categories are not too broad and general. The generality of category prevents fine-grain classification and can lead to ambiguities [15]. The classification of this paper is unambiguous.

NodeOS classifies core and interface part. The core part includes Domains, Thread Pool, Memory Pool, Channels, and Files. And interface part includes with the rest part except core part: events, heap, packets, time, and so forth. EE classifies preload and postload part. If EE does not load, there is no the vulnerability in EE. Therefore, the criterion of classification is suitable. AP classifies code base and application base part. In this paper, AP rename instead of AA. AA means AP that have AN's resource. As a packet that AP contains AC and data, code base is a previous status to occupy the resources of AN. application base refers to a status that is occupying resources of AN. Status of AP can do that is important criterion and is good feature in such a side that performance of AN. Classification criteria of component feature domains show as Table 1.

Table 1. Classification criteria of components dependent domain

Active components	Classification criteria
NodeOS	System feature
EE	Precedence feature
AP	Resource feature

We analyzed active network components and classified domains. In order to classify vulnerability domains, we describe exceptions of components [9]. For example, Exceptions that can be happened in NodeOS are as follows.

Table 2. Classification of security vulnerability in active network environments

Infrastructure	Components	Components dependent (security label)	Vulnerability	Exception examples
Active networks	NodeOS	Interface(Low)	environment exception	EE loading exception
			boundary condition exception	AA revoke exception
			fail to handle exception conditions	binding exception
		Core(High)	design exception	user code terminated exception
			configuration exception	weak permission setting exception of file
			active boundary exception	in/out/cut channel exception
			active component handling exception	OSVM exception
	EE	PreLoad(Low)	environment exception	EE is not supported code execution
			configuration exception	EE naming exception
		PostLoad(High)	design exception	data casting exception
			environment exception	interaction exception that other modules
			configuration exception	setting weak permission exception
			input validation exception	exception that EE could allow for the arbitrary execution of commands
			boundary condition exception	rate limitation exception
			active component handling exception	AP handling exception
			recursive exception	EE interruption exception
	AP	Code base(Low)	access validation exception	AC authentication exception
			boundary condition exception	rate limitation rule exception
			code check exception	PCC type check exception
		Appl. base(High)	origin exception	exception that AA executed with being not authenticated
			boundary condition exception	code terminate exception
			active component handling exception	crash that malicious code is cause
			recursive exception	exception that disturb each other AA

Binding exception: OS virtual machine approach NodeOS interface through NodeOS binding. At this process, Exception can occur.

User code terminated exception: User code should be terminates compulsorily. At this process, Exception can occur.

In/out/cut channel exception: Because incoming/cut-through/outgoing channels exception, processing flow can be wrong.

To be precise, more potential exceptions can occur. Exceptions of NodeOS can be removed through update, patch, and reconfiguration.

EE can be potential exceptions such as data casting, security bypass, and calling mistaken between EEs. Exceptions that can be happened in EE are as follows.

Data casting exception: When EE is running AA, exception about data casting can occur.

Security bypass exception; Exception can happen through security bypass about weak configuration.

EE interruption exception: AN supports multiplex EE. At this process, exception can be interrupt between EEs.

Above exceptions of EE can be removed by enforcing EE's security mechanism.

AP can be exceptions such as Denial of Service, disturbance of other AA, resources destruction of node, and sensitive information leakage, and so forth [14]. Exceptions that can be happened in AP are as follows.

Access violation exception of AA: Exception that approach other user's application can happen.

Exception about excessive resources use of AA: Regardless of limited resources, Exception can be executed AA.

PCC(Proof Carrying Code) type check exception: Exceptions by wrong proof algorithm(or code) of PCC can occur.

Above exceptions of AP can be removed through checking fingerprint in code based active packet, using rate limitation rule in application based active packet.

We categorized vulnerability domains based on above exceptions using reverse engineering. Reverse engineering means that refers to re-creation work about traditional vulnerability. For example, in the case of NodeOS, we analyzed redhat, caldera, debian, slackware, turbo linux, netBSD, freeBSD, Solaris, and so forth. We classified EE and AP by same method. In this step, we propose new vulnerability factors such as active component handling, recursive exception, and code check exception. There are no these factors in traditional category. Our analysis can be classified into Table 2.

6 Conclusion and Future Works

This paper describes taxonomy of security vulnerability based on active network environments. We attempted to classify vulnerability in active network environments for the first time. When vulnerability occurs, this taxonomy clarifies

whether happened from some components. Also, vulnerability level in component dependent domain can enforces policy of vulnerability scanning.

It needs to be evaluated whether our taxonomy needs to be enhanced to encompass active network environments. Therefore, we may construct an active network environments for testing of described vulnerabilities. We will test vulnerabilities about NodeOS and EEs and APs in the future works and study more detail policy of vulnerability scanning using security labeling.

References

1. A. Silberschatz, et al., "Operating system concepts", 6th Ed., John wiley and sons, inc., 2002.
2. AN NodeOS Working Group, "NodeOS Interface Specification", Nov. 2001.
3. AN Security Working Group, "Security Architecture for Active Nets", Nov. 2001.
4. D. Raz and Y. Shavitt, "An Active Network Approach to Efficient Network Management", IWAN'99, 1999.
5. D. J. Wetherall, et al., "ANTS: A Toolkit for Building and Dynamically Deploying Network Protocols", IEEE OPENARCH'98 Proc., San Francisco, Apr. 1998.
6. D. L. Tennenhouse, et al., "A Survey of Active Network Research", IEEE communications magazine, pp.80 86, Jan. 1997.
7. D. L. Tennenhouse and D. J. Wetherall, "Towards an active network architecture", In Multimedia Computing and Networking'96, Jan. 1996.
8. H. K. Kim, et al., "Vulnerability Management Architecture for Active Nodes", KNOM Review Vol. 5, No. 2, Dec. 2002.
9. J. S. Yang, et al., "A Study on Security Vulnerability of Active Network", Proc. of the 19th KIPS Fall Conference, Korea, May 2003.
10. K. Psounis, "Active Networks: Applications, Security, Safety and Architectures", IEEE Communications Surveys, First Quarter 1999.
11. K. Calvert, "Architectural Framework for Active Networks. Technical report", AN Architecture Working Group, 2000.
12. M. Hicks, et al., "PLAN: A Packet Language for Active Networks", ICFP, 1998.
13. P. Tullmann, et al., "JANOS: A Java-Oriented OS for Active Network Nodes", IEEE JOURNAL ON SELECTED AREAS IN COMMUNICAT IONS, Vol. 19, No. 3, Mar. 2001.
14. S. Oaks, Java Security, O'REILLY, Jun. 2001.
15. T. Aslam, et al., "Use of a Taxonomy of Security faults", Proc. of the national computer security conference;Coast laboratory Technical Report 96-05; 1996.
16. Certcc-kr Homepage, http://www.certcc.or.kr/
17. Security Focus Homepage, http://www.securityfocus.com/

A Secure and Flexible Multi-signcryption Scheme

Seung-Hyun Seo and Sang-Ho Lee

Department of Computer Science and Engineering,
Ewha Womans University,
11-1 Daehyun-dong, Seodaemun-ku, Seoul 120-750, Korea
{happyday, shlee}@ewha.ac.kr

Abstract. Multi-signcryption scheme is an extension of signcryption scheme for multi-signers performing together the signcryption operation on messages, and it provides useful cryptographic functions such as confidentiality and authenticity for the sound circulation of messages through the Internet. In this paper, we show the weaknesses of the previous multi-signcryption schemes. And then we propose a new multi-signcryption scheme that improves the weaknesses and the efficiency of the previous schemes. Our scheme efficiently provides message flexibility, order flexibility, message verifiability, order verifiability, message confidentiality, message unforgeability, non-repudiation and robustness. Therefore, it is suitable for protecting messages and multi-signers from malicious attacks in the Internet.

Keywords. Signcryption, multi-signature.

1 Introduction

With the continuous growth of the Internet, the messages such as data, software program or documents have been circulated through the Internet. In such environment, the Internet user sends and forwards an original message to other users. Through this process, the message may be modified, improved and added a convenient feature by many users. But, since malicious attackers can insert the malicious code such as computer virus into the circulating message or wrongly modify the original message, we must detect the malicious attackers and prevent the malicious code from damaging the receiver[1,2,3]. Especially, if the modified part of the original message is private thing then we must prevent the attackers from obtaining the modified messages. Moreover, since the copyright problem has been often happened, we must be able to distinguish an original author from other authors who modify the original message. For these reasons, the multi-signcryption scheme should be required.

The multi-signcryption scheme is a cryptographic method that fulfills both the functions of secure encryption and digital multi-signature[1,2] for multi-users. Since the multi-signcryption scheme provides message confidentiality, authenticity and robustness, it is suitable for such environments.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 689–697, 2004.
© Springer-Verlag Berlin Heidelberg 2004

Recently, Mitomi and Miyaji proposed the multi-signcryption scheme[3] with message flexibility, order flexibility, message verifiability, order verifiability and robustness. But, since their scheme does not provide the message confidentiality, it cannot prevent a malicious attacker from obtaining the information on the messages. Pang, Catania and Tan proposed the modified multi-signcryption scheme to achieve the message confidentiality[5]. However, their scheme does not satisfy the order flexibility and requires a lot of modular multiplications.

In this paper, we propose a new secure and flexible multi-signcryption scheme. Our scheme solves the weaknesses of the previous schemes, and provides message flexibility, order flexibility, message verifiability, order verifiability, message confidentiality, message unforgeability, non-repudiation and robustness. In addition, our scheme is more efficient than the previous schemes.

The rest of the paper is organized as follows. In section 2, we describe the definitions and the requirements for multi- signcryption. In section 3, we give brief description of Mitomi-Miyaji scheme and Pang-Catania-Tan scheme, and explain their weaknesses. In section 4, we present our secure and flexible multi-signcryption scheme. In section 5, we analyze the features, the security and the efficiency of our scheme. Finally, we draw our conclusions in section 6.

2 Definitions and Requirements of Multi-signcryption

In this section we describe the definitions and the requirements of the multi-signcryption scheme. Multi-signcryption scheme is an extension of signcryption scheme for multi-users performing together the signcryption operation on messages[5]. In other words, the multi-signcryption scheme is a cryptographic method that fulfills both the functions of secure encryption and digital multi-signature for multi-users, but with a cost smaller than that required by multi-signature-then-encryption[6,7].

To construct a secure and flexible multi-signcryption scheme, the following requirements must be satisfied[3,6,7];

1. **Message flexibility:** A message does not need to be fixed beforehand.
2. **Order flexibility:** Order of signers does not need to be designated beforehand.
3. **Message and order verifiability:** A verifier can verify who is an original author of a message, who modifies an original message and furthermore to which order a message is modified.
4. **Message confidentiality:** It is computationally infeasible for an attacker (who may be any dishonest entity other than verifier) to gain any partial information on the contents of a multi-signctyped message.
5. **Message unforgeability:** It is computationally infeasible for an attacker (who may be dishonest entity or verifier) to masquerade signers in creating a multi-signcrypted message.
6. **Non-repudiation:** Signers cannot falsely deny later that he generated a multi-signcryted message.

7. **Robustness:** If the multi-signature verification on a message fails, then unauthentic encrypted messages cannot be decrypted. So, it prevent such the unauthentic messages from damaging a receiver.

3 Previous Work

In this section, we summarize the previous multi-signcrytion schemes and describe their weaknesses. The notations for this paper are as follows: I_j is the j-th signer($1 \leq j \leq n$) who generates the multi-signcryption on the modified message, I_0 is an original signer who generates signature on the original message, V is a verifier and ID_i is the identity information of $I_i (0 \leq i \leq n)$. The system parameters consist of a large prime p, a prime factor q of $p-1$, and a generator g for Z_p^*. The signer $I_i (0 \leq i \leq n)$'s secret key is $x_i \in Z_q^*$ and the corresponding public key is $y_i = g^{x_i} \pmod p$. $H(\cdot)$, $h_1(\cdot)$ and $h_2(\cdot)$ are strong one-way hash functions, and $E_K(\cdot)$ is a symmetric encryption function with key K.

3.1 Mitomi-Miyaji Scheme

Mitomi and Miyaji proposed multi-signcryption scheme which combines their multi-signature with the encryption function[3]. Their scheme is shown in the Figure 1. After the first signer I_1 generates the multi-signcryption on m_1, he sends the multi-signcrypted message(ID_1, s_1, C_1) to the next signer I_j. I_j verifies and decrypts the message, and he creates m_j that modifies m_1, and he generates the multi-signcryption (ID_j, s_j, C_j) on the m_j, and then he sends(ID_j, s_j, C_j) to the next signer. Finally the last signer I_n sends the multi-signcrypted messages (ID_1, s_1, C_1), $\cdots$, (ID_n, s_n, r_n, C_n) to the verifier V.

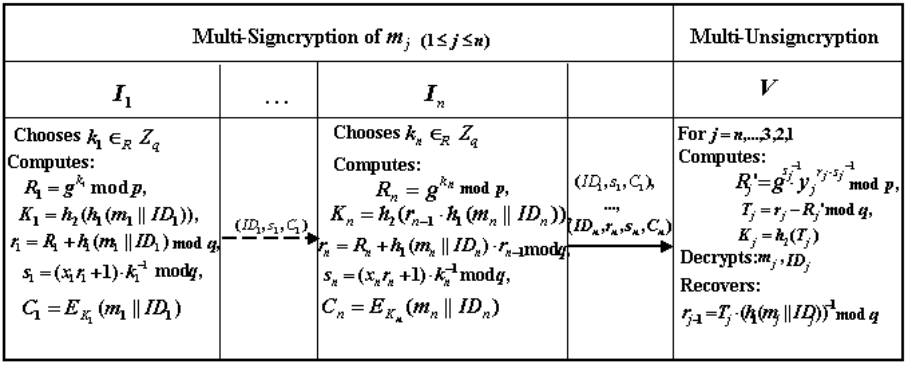


Fig. 1. Mitomi-Miyaji Scheme

This multi-signcryption scheme provides the robustness that message cannot be recovered if the signature verification fails[3]. But, since the multi-signcrypted

messages can be easily verified that $R_j = g^{s_j^{-1}} \cdot y_j^{r_j \cdot s_n^{-1}}$ for $j = n, ..., 1$ by using the signer's public key y_j and the received values (s_j, r_j)(,where r_n is received value and r_{j-1} is recovered from r_j for $j = n, ..., 2$), anyone can verify the multi-signcrypted message and generate the session key $K_j = h_2(r_j - R_j)$ So, anyone can decrypt the multi-signcrypted messages by using the session key. Therefore, Mitomi-Miyaji scheme cannot provide the message confidentiality. If the modified messages requires to be securely protected from the attacker, this scheme is not suitable.

3.2 Pang-Catania-Tan Scheme

Pang, Catania, and Tan[5] modified the Mitomi-Miyaji multi-signcryption scheme to satisfy the message confidentiality. Their scheme is shown in the Figure 2. After the original signer I_0 determines the order of multi-signers and generates the signature (ID_0, s_0, r_0) on m_0, he sends (ID_0, s_0, r_0) to the first signer I_1. After I_1 creates m_1 that modifies m_0, he generates the multi-signcryption (ID_1, s_1, C_1) on the m_1, and then he sends(ID_1, s_1, C_1) to the next signer. Finally the last signer I_n sends the multi-signcrypted messages (ID_1, s_1, C_1), (ID_2, s_2, C_2), $\cdots$, (ID_n, s_n, r_n, C_n) to the original signer I_0.

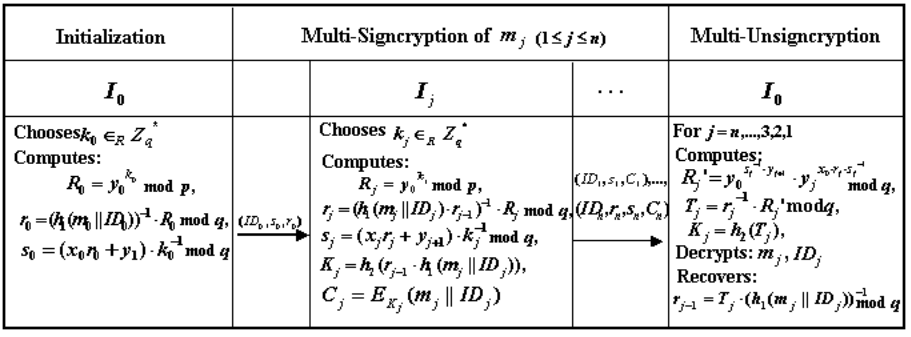


Fig. 2. Pang-Catania-Tan Scheme

Unlike Mitomi-Miyaji scheme,this scheme permits only the original signer I_0 to verify and decrypt the multi-signcrypted messages. Since the verification process requires the I_0's secret key x_0, other ones cannot verify and decrypt the multi-signcrypted messages. So, this scheme provides the message confidentiality. But, since the I_0 fixes the order of multi-signers $I_j (1 \le j \le n)$beforehand, it does not provide the order flexibility. So,after setting up the order of multi-signers for the multi-signcryption, a signer can be neither added nor excluded and order of signers cannot be changed.

4 Proposed Multi-signcryption Scheme

In this section, we propose a new multi-signcryption scheme that resolves the weaknesses of the Mitomi-Miyaji scheme and the Pang-Catania-Tan scheme. Unlike the previous schemes, Our scheme satisfies not only the message confidentiality and the order flexibility but also any other requirements for the secure and flexible multi-signcryption. Moreover,our scheme is more efficient than the previous schemes. We mention the detailed reasons in the next section.

Our scheme is composed of two phases such as multi-signcryption phase and multi-unsigncryption phase. In our scheme, the original signer's secret key x_0 should be required to verify the multi-signcrypted messages and decrypt them. So, only the original signer I_0 can verify and decrypt them. Figure 3 shows our scheme. After the I_0 publishes his public key $y_0 = g^{k_0} \pmod p$ and original message m_0 to multi-signers, the proposed scheme proceeds with as follows:

[Multi-Signcryption phase] In this Multi-Signcryption phase, multi-signers generate multi-signcryption on messages.

1. The first signer I_1 chooses random $k_1 \in_R Z_q^*$ and computes session key $K_1 = y_0^{k_1} \pmod p$ for the multi-signcryption. And then he creates the message m_1 by modifying m_0 and generates multi-signcryption on m_1 as follows: $r_1 = H(m_1 \| ID_1 \| K_1) \pmod q$, $s_1 = (x_1 + r_1) \cdot k_1^{-1} \pmod q$, $C_1 = E_{K_1}(m_1 \| ID_1)$, where the multi-signcrypted message is composed of the multi-signature (r_1, s_1) to authenticate the multi-signer and the encrypted message C_1 to protect the m_1. He sends multi-signcrypted message (ID_1, s_1, r_1, C_1) to the next signer I_j.
   In our scheme, since the order of signers is not designated beforehand, anyone can participate the multi-signcryption. For convenience, we denote I_j's next signer by I_{j+1}.
2. The signer $I_j (2 \leq j \leq n)$ chooses random $k_j \in_R Z_q^*$ and computes session key $K_j = y_0^{k_j} \pmod p$. And then he creates the message m_j by modifying m_0, and generates multi-signcryption on the message m_j as follows: $r_j = H(m_j \| ID_j \| K_j) \cdot r_{j-1} \pmod q$, $s_j = (x_j + r_j) \cdot k_j^{-1} \pmod q$, $C_j = E_{K_j}(m_j \| ID_j \| ID_{j-1} \| s_{j-1} \| C_{j-1})$,where the multi-signcrypted message is composed of the multi-signature (r_j, s_j) to authenticate the multi-signer and the encrypted message C_j to protect the m_j. If I_j is the final signer$(I_j = I_n)$, then he sends (ID_n, s_n, r_n, C_n) to the verifier. Otherwise, I_j sends (ID_j, s_j, r_j, C_j) to the next signer I_{j+1}.

[Multi-Unsigncryption phase] In this Multi-Unsigncryption phase, the original signer verifies the multi-signature(r_j, s_j) and decrypts theC_j.

1. The original signer I_0 receives the multi-signcrypted messages (ID_j, s_j, r_j, C_j) from I_j.
2. For $j = n, \cdots, 3, 2, 1$, V computes session key K'_j using his private key x_0 as follows:

$$K'_j = (y_j \cdot g^{r_j})^{s_j^{-1} \cdot x_0} = g^{(x_j + r_j) \cdot (x_j + r_j)^{-1} \cdot k_j \cdot x_0} = g^{k_j \cdot x_0} \pmod p$$

If $K'_j = K_j$, then he can decrypt $m_j, ID_j, ID_{j-1}, s_{j-1}$, and $C_j - 1$ correctly and know the I_j's previous signer I_{j-1} and a part of his signature s_{j-1}. After he decrypts them, he recovers $r_{j-1} = H(m_j||ID_j||K_j)^{-1} \cdot r_j \pmod q$. Let $j = j - 1$.

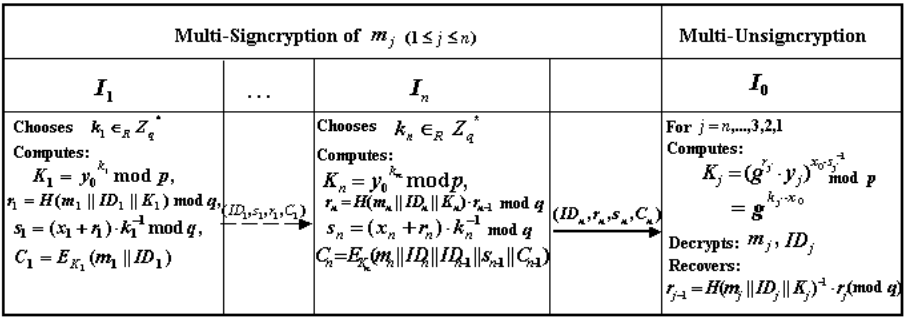


Fig. 3. Proposed Multi-Signcryption Scheme

5 Analysis of the Proposed Multi-signcryption Scheme

In this section, we analyze the features, the security and the efficiency of our scheme. Table 1 shows the features and security comparisons with original signcryption scheme, Mitomi-Miyaji scheme, Pang-Catania-Tan scheme and our scheme. Table 2 shows the efficiency comparisons with four schemes. Here, the original signcryption scheme implies a simple chain of Zheng's signcryption[6,7]. I.e., each signer generates the signcrypted message and sends it together with the previous signer's signcrypted message.

5.1 Feature and Security Analysis

Our scheme satisfies all requirements for secure and flexible multi-signcryption as follows:

1. **Message flexibility:** In our scheme, the message is not fixed beforehand. So, each signer can generate multi-signcryption on the message.
2. **Order flexibility:**In our scheme, the order of signers is not designated beforehand. So, each signer can determine the next signer, and signers can easily change the order of signers.
3. **Message and order verifiability:** Since the verifier(original signer I_0) receives the multi-signcrypted message with the last signer's ID, he can confirm the last signer. And he can know the rest of signers by decrypting the $C_n, C_{n-1}, ..., C_1$. So, he can know the order of signers and verify who is the signer of the multi-signcrypted message by using the signer's public key.

Table 1. Feature and Security comparisions

	Zheng	Mitomi-Miyaji	Pang-Catania-Tan	Ours
Message Flexibility	○	○	○	○
Order Flexibility	○	○	×	○
Message Verifiability	○	○	○	○
Order Verifiability	×	○	○	○
Message Confidentiality	○	×	○	○
Message Unforgeability	○	○	○	○
Non-Repudiation	○	○	○	○
Robustness	○	○	○	○

4. **Message confidentiality:** If an attacker eavesdrops the communication between the signers and the verifier(original signer I_0), then he can obtain the multi-signcrypted messages (ID_1, s_1, r_1, C_1), (ID_2, s_2, r_2, C_2), ..., (ID_n, s_n, r_n, C_n) of messages m_1, m_2, ..., m_n. And the attacker can compute $(y_j \cdot g^{r_j})^{s_j^{-1}}\ modp = g^{k_j} mod\ p (1 \leq j \leq n)$ from the multi-signcrypted messages. But, since the attacker cannot know the verifier(I_0)'s private key, he cannot compute session keys as the difficulty of discrete logarithm problem[4]. Therefore, it is computationally infeasible for the attacker to gain any information on the modified messages $m_1, m_2, ..., m_n$.
5. **Message unforgeability:** Let assume that the dishonest verifier(I_0) and the attacker try to forge the signerI_j's $(1 \leq j \leq n)$ multi-signcrpytion.
   First, in the case of the dishonest verifier, he tries to send the forged multi-signcrypted message $(ID_j, s_j^{'}, r_j^{'}, C_j^{'})$on the forged message $m_j^{'}$ to the other verifier V. He may use his private key x_0 to decrypt the I_j's multi-signcryption and obtain the message m_j. And then he forges m_j into $m_j^{'}$, chooses $k_j^{'} \in_R {Z_q}^*$, computes session key $K_j^{'} = {y_{v^*}}^{k_j^{'}} modp$, where the V's public key is y_v, and creates $r_j^{'} = H(m_j \| ID_j \| K_j^{'}) \cdot r_{j-1} modq$. But he cannot generate $s_j^{'} = (x_j + r_j^{'}) \cdot {k_j^{'}}^{-1} modq$, because she does not know the I_j's private key x_j. So, the dishonest verifier(I_0) cannot forge the signer I_j's multi-signcryption.
   In the other case, the attacker tries to send the multi-signcrypted message $(ID_j, s_j^{"}, r_j^{"}, C_j^{"})$on the forged message $m_j^{"}$ to the verifier(I_0) by masquerading as the I_j. But, even though he can compute $K_j^{"} = {y_0}^{k_j^{"}} modp$ and $r_j^{"} = H(m_j^{"} \| ID_j \| K_j^{"}) \cdot r_{j-1} modq$ arbitrarily, since he does not know the I_j's private key x_j, he cannot generate $s_j^{"}$. Therefore, he cannot forge the I_j's multi-signcryption.
6. **Non-repudiation:** Since each multi-signcrypted message includes the signer I_j's$(1 \leq j \leq n)$private key, anyone who does not know the private key cannot generate multi-signcrypted message instead of I_j. Therefore, if once I_j generates the multi-signcrypted message, he cannot falsely deny later the fact that he generated it.

7. **Robustness:**If the verification $K_j^{'} = (y_j \cdot g^{r_j})^{s_j^{-1} \cdot x_0} modp$ of the multi-signcryption fails, then the verifier I_0 cannot compute the session key K_j. So, since he cannot decrypt the cipertext C_j, he can prevent the damage by the unauthentic message. Therefore, our scheme provides the robustness.

5.2 Efficiency Analysis

We evaluate our multi-signcryption scheme from a point of view of computational cost and communication overhead. We use the number of modular multiplications and the number of exponentiations to measure the computational cost.

For convenience, we assume the following conditions: (1) we denote the number of signers by n, the computational amount for a N-bit modular multiplication by m_N, the computational amount for a N-bit modular exponentiation by e_N and the one message size by $|M|$bits; (2) two prime p and q are set to 1024 and 160 bits respectively; (3) the output size of the cryptographic hash functions is 160 bits.

We compare our scheme with the original signcryption scheme, the Mitomi-Miyaji scheme and the Pang-Catania-Tan scheme. Table 2 shows the efficiency comparisons of four schemes. The computational amount of our scheme is lower than that of Mitomi-Miyaji and Pang-Catania-Tan, and moreover our scheme satisfies all requirements for the multi-signcryption.

The communication overhead of Mitomi-Miyaji scheme, the Pang-Catania-Tan scheme and our scheme is $n \cdot |M| + (n+1) \cdot |q| = n \cdot (|M| + 160) + 160$. But, in comparison with the Pang-Catania-Tan scheme, our scheme reduces the verifier's computational cost of 160-bit modular multiplications from $(4n + 2) \cdot m_{160}$to $2n \cdot m_{160}$. It means 50% reduction in average verifier's computational cost of the multiplications. And it reduces the signer's computational cost of 160-bit modular multiplications from $4 \cdot m_{160}$ to $2 \cdot m_{160}$. This represents 50% reduction in average signer's computational cost of the 160-bit modular multiplications. Moreover, in comparison with the Mitomi-Miyaji scheme, our scheme reduces the signer's computational cost of 160-bit modular multiplications from $3 \cdot m_{160}$ to $2 \cdot m_{160}$. It means 33% reduction in average signer's computational cost of the 160-bit modular multiplications.

In comparison with the original signcryption scheme, our scheme has the tradeoff between the computational amount and the communication overhead.

Table 2. Efficiency comparisions

	Zheng		Mitomi-Miyaji		Pang-Catania-Tan		Ours	
	signer	verifier	signer	verifier	signer	verifier	signer	verifier
160-bit Mod Mul.	1	n	3	$2n$	4	$4n+2$	2	$2n$
1024-bit Mod Mul.	0	n	0	n	0	n	0	n
1024-bit Mod Expo. ($\|q\| = 160$)	1	$2n$	1	$2n$	1	$2n+1$	1	$2n$

So,though our scheme increases the verifier's computational cost of 160-bit modular multiplications more than the original signcryption scheme, it reduces the communication overhead to the most 50%. (Here,the communication overhead of the original signcryption scheme is $n \cdot |M| + n \cdot |q| + n \cdot |H(.)| = n \cdot (|M| + 320)$, and that of our scheme is $n \cdot (|M| + 160) + 160$.)

6 Conclusions

The multi-signcryption scheme is necessary for many Internet users to protect the circulating messages from malicious attacks, and to authenticate each signer who modifies the original message.

In this paper, we proposed a new secure and flexible multi-signcryption scheme that improves of the previous schemes[2,3,5]. Our scheme satisfies the message flexibility, the order flexibility, the message verifiability, the order verifiability, the message confidentiality, the message unforgeability, the non-repudiation and the robustness. Moreover, since it has much less computational cost than the previous schemes, we expect that our scheme can be used to circulate messages efficiently through the Internet.

References

1. M. Burmester, Yvo Desmedt, H. Doi, M. Mambo, E. Okamoto, M. Tada, and Y. Yoshifuji: *A Structed ElGamal-Type Multisignature Scheme*, Advances in Cryptology- In Proceedings of PKC 2000, LNCS, Springer-Verlag 2000, pages 466-482.
2. S. Mitomi and A. Miyaji: *A Multisignature Schemes with Message Flexibility, Order Flexibility, and Order Verifiability*, In Proceedings of ACISP 2000, Vol.1841 of LNCS, Springer-Verlag 2000, pages 298-312.
3. S. Mitomi and A. Miyaji: *A General Model of Multisignature Schemes with Message Flexibility, Order Flexibility, and Order Verifiability*, IEICE Transaction on Fundamentals, Vol. E84-A, No. 10, 2001, pages 2488-2499.
4. A. J. Menezes, P. C. Oorschot and S. A. Vanstone: *Handbook of Applied Cryptography*, CRC, 1997.
5. X.Pang, B.Catania, and K-L, Tan: *Securing Your Data in Agent-Based P2P Systems*, In Proceedings of Eight International Conference on Database Systems for Advanced Applications(DASFAA '03), 2003.
6. Y. Zheng: *Signcryption and Its Applications in Efficient public Key Solutions*, In Proceedings of 1997 Information Security Workshop (ISW'97), Vol.1397 of LNCS, Springer-Verlag, 1997, pages 291-312.
7. Y. Zheng:*Digital Signcryption or How to Achieve Cost (Signature & Encryption)* $\ll$ *Cost (Signature) + Cost (Encryption)*, Advances in Cryptology - Crypto'97, Vol. 1294 of LNCS, Springer-Verlag, 1997, pages 165-179.

User Authentication Protocol Based on Human Memorable Password and Using RSA

IkSu Park[1], SeungBae Park[2], and ByeongKyun Oh[3]

[1] Dept. of Information Security, Mokpo National University, Muan-gun Jeonnam, KOREA
ispark@mis2.mokpo.ac.kr

[2] Dept. of Computer Science, Chodang University, Muan-gun Jeonnam, KOREA
sbpark@chodang.ac.kr

[3] Dept. of Information Security, Mokpo National University, Muan-gun Jeonnam, KOREA
obk@mokpo.ac.kr

Abstract. Until now, authentication protocol using the suggested password is not safe from off-line dictionary attack or password file compromise. On this paper, we define scheme password based authentication protocol (PAP) authentication protocol using password. PAP features managing one value choosing optionally of expressing password of many values. It presents PAP based authentication protocol, PAPRSA using RSA to manage values expressing password. PAPRSA is safe from attack involving off-line dictionary attack, password file compromise and excellent in efficient ways involving pass number, calculation amount.

Keywords: Authentication, Cryptography, Password authentication, Password Dictionary, public key cryptography

1 Introduction

Authentication is procedure to confirm whether it is apply used or not, authentication protocol is authenticating protocol keep secret information safe on communication-line and server. There are password based formally, challenge response protocol, zero- knowledge protocol on presented authentication protocol by this till.

Password based authentication protocol uses one-way function or salt of UNIX [5, 10]. OTP (one-time password) is password based formally using one-way function, so prover sends from one-way function which is sent message to verifier the result applying one more time. Although OTP is safe to replay attack because the message sent to verifier is different, it is not safe from pre-play, off-line dictionary attack, server compromise [21].

There are formally using symmetric key encryption algorithm, public key cryptosystem in challenge response protocol [13, 14]. The verifier who uses symmetric key encryption algorithm in challenge response protocol sends challenge (randomly and secretly) to prover, and verifier repeats procedure sending response corresponding about challenge to verifier. Symmetric key encryption algorithm based authentication protocol has fast speed about one transaction, however it happens key management

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 698–707, 2004.
© Springer-Verlag Berlin Heidelberg 2004

problems. And the user also uses difficult key to memorize, it needs addition hard to memory to store key. The verifier who uses public key cryptosystem in challenge response protocol sends challenge to prover, and verifier repeats procedure sending response corresponding about challenge to verifier [6, 16, 18]. As public key cryptosystem based authentication protocol uses public key cryptosystem, management of key is easy, but the speed is slow. And the user also user difficult key to memorize, it needs addition hard to memory to store key.

The verifier in Zero knowledge protocol is a formally to attempt to ask many questions to prover [7]. By answering correctly to all answers to attempt, he checks he is a right verifier. While the verifier does not expose the information having itself to verifier in zero knowledge authentication protocol, although the can check the fact that he has information to verifier, because of many pass value, the speed of procedure is slow, the verifier user difficult information to memorize, so it needs addition hard to memory in order to store information [7, 8, 11].

From authentication protocol presented, 1) pass value calculation amount are little, 2) it is safe from authentication protocol attack like pre-play attack, off-line dictionary attack, password file compromise, 3) we can know to store secret information without sending authentication protocol.

This paper, we define scheme password based authentication protocol (PAP), authentication protocol using password. Optionally, although the user is possible password to memorize, of users that the number of expressing password can be endless PAP features managing one value choosing optionally of expression password of many values.

In PAP based authentication protocol, attacker 1) Can try to know password analyzing public key cryptosystem or 2) Attempt to know password dictionary attack using password dictionary, complete attack. About PAP based authentication protocol, we define powerfully secure, safety of public key cryptosystem and the meaning of without analyzing public key cryptosystem from considering safety about attack to analogize password from message eavesdropping.

We present PAP based authentication protocol, PAPRSA using RSA to treat values expressing password. PAPRSA is powerfully secure from pre-play attack, off-line dictionary attack, safe from password file compromise.

The pass value of PAPRSA is one, superior in efficiency ways. The organization of this paper is like that. In chapter 3, we define PAP, in chapter 4, we suggest PAPRSA.

2 Attack about Authentication Protocol

Attack about authentication protocol is absolute eavesdropping.

1. Replay: It is attack that attacker impersonation with right using itself the message acquired during communication.

2. Pre-play: Attacker determines next communication message using current communication message, and it is attack to impersonation to right user using determined message.
3. Man-in-the-middle: it is attack that it impersonation with verifier communication message sent to prover and prover to verifier. In the main, it is applied to mutual authentication protocol or protocol which is fixed with pass value.
4. Password guessing attacks: Attackers basically use dictionary collection something that can be possible password, it is divided into on-line dictionary attack, off-line dictionary attack.

Off-line dictionary attack compare with value in password dictionary interrupt communication message among user, is impersonation attack to user using password driven corresponded value [1, 2, 3, 4, 9, 10]. Using off-line dictionary attack, attacker can be used dictionary password, the result treated password dictionary attack was storing in the dictionary.

On-line dictionary attack attacker repeat the password in the dictionary until appearing the term of validity password attempting process choosing one by one. Now, On-line dictionary attack set password possible that we do not consider on-line dictionary attack on this paper [12, 15].

5. Server Compromise: Attacker basically impersonation using secret information to sever or attacker may impersonation to user [12, 17, 19, 21].

3 PAP: Password Authentication Protocol Scheme

PAP is authentication protocol scheme composed with registering procedure, authenticating procedure.

PROTOCOL

Registering procedure

Input: an identity *id* and a user is password *P*'

1. Get public key and private key of a public key of a public key cryptosystem, and then publishes the public key;
2. Chooses x_1 in random and then determines x_2 in an equation of $x_1 \& x_2 = P$ or $x_1 \&_1 x_2 \&_2 x_3 = P$; (where &, $\&_1$, and $\&_2$ denote the operators, and x_3 is a shared value between the prover and the verifier)
3. A cryptographic function or algorithm maps x_1 to another value in default and x_2 in option;
4. Let *F* be the cryptographic function or algorithm used in 3, then stores ($F(x_1)$, x_2), or ($F(x_1)$, $F(x_2)$) at *id*.

Authentication procedure

Input: an identity *id* and *P*' inputted by the user as *P*.

1. A prover chooses y_1 in random and then determines y_2 in an equation of $y_1 \& y_2 = P$' or $y_1 \&_1 y_2 \&_2 y_3 = P$';
2. The prover encrypts y_1 by using the public key of the verifier in default and y_2 in option;

3. At 2, If in case y_2 was encrypted by the prover, the prover sends two ciphertexts for verifier and if in case of only y_1 encrypted, the prover sends y_2 correspondence in y_1 ciphertexts for verifier;
4. A verifier to determine whether *P*=*P*' or not, compares two information.

On PAP, driven value from password is (x_1, x_2), (y_1, y_2). And (x_1, x_2) or x_1 is encryption by public-key cryptosystem be for sending to verifier. Therefore, to know password after eavesdropping message, attacker analyze public key cryptosystem or must attempt like complete attack about password dictionary attack or perfect forward attack. Because of this reason, on PAP based authentication protocol, we must consider in case know password analyzing public key cryptosystem and safety about attack to inference password from message eavesdropping without analyzing public key cryptosystem at the same time.

Definition 1. About attack, using message eavesdropping, if safety of public key cryptosystem, we call PAP based authenticating protocol powerfully secure.

4 PAPRSA

On this section, we presents of PAP using RSA.

4.1 Notation

id: Identity of a user.
P: A genuine password.
P': A password inputted when a user accesses to the verifier.
p, *q*: Two primes suitable for RSA.
N: $N = pq$.
Ø(*n*): $(p-1)(q-1)$.
e: An integer that is relatively prime to Ø(*n*).
d: $ed \equiv 1 \bmod Ø(n)$.
Z: $\{1, 2, \ldots, N-1\}$.
x_1, x_2, y_1, y_2: Elements of *Z*.
t: Timestamp.

4.2 Protocol

PAPRSA is described in Fig. 1.

PROTOCOL
Registering procedure
Input: *id*, *P*.

1. Get $N=pq$ and (e, d);
2. Publish e and N;
3. Chooses y_1 in Z in random and then determine y_2 so that y_1-y_2=P;
4. Get $y_1^2 \bmod N$ and $y_2^2 \bmod N$, and then stores them at id.

Authentication procedure

Input: id, P.

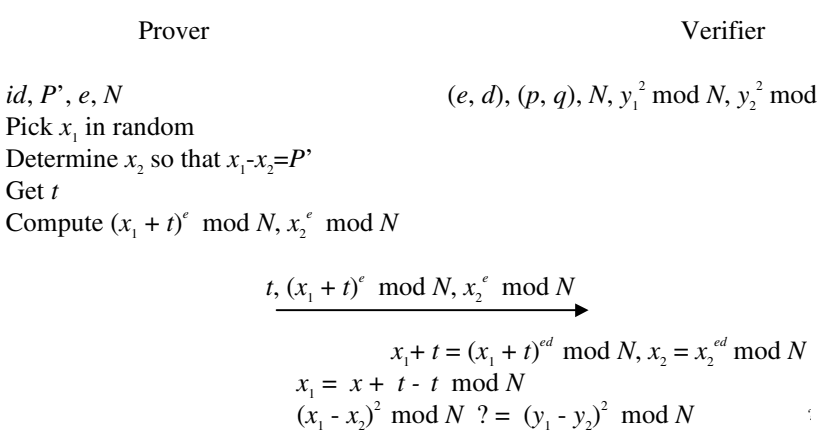


Fig. 1. PAPRSA

Lemma 1. The verifier in the PAPRSA is feasible determine correctly whether it is authorized the prover or not.

Proof: The verifier can obtain (x_1, x_2) by decryption using private key from reception the prover ciphertext. The verifier know that $(x_1, x_2) = (y_1, y_2)$. Therefore, the verifier is feasible determine whether it is currently system approach authorized the prover or not from inspected $(x_1 - x_2)^2 \bmod N = (y_1 - y_2)^2 \bmod N$.

5 PAPRSA Analysis

On this subsection, we analyze safety of PAPRSA and efficient.

5.1 Safety

Safety in the PAPRSA, of expressing password of values number using fact selected in random in the Z candidates and safe selected values for expressing password, depends on the security of RSA.

Theorem 2. PAPRSA is powerfully secure from pre-play attack, off-line dictionary attack, man-in-the-middle attack.

Proof: Let, $S=\{(z_1^i, z_2^i) \mid 1< i < N\text{-}1, z_1^i + z_2^i = P\}$. Then (x_1, x_2) is in S and because of x_1 is value selected random in the set theory j $\mid 1 < i < N$-1, (x_1, x_2) is a pair ordered perform two number of random choose in S. If attacker was eavesdropping communication message from prover, for impersonation attacker has feasible determine (x_1, x_2) from encryption cipertext using RSA. However, in the RSA, space for determine p and q is small for n-1, attacker has space feasible determine (x_1, x_2) is big for space for determine p and q in the RSA [19].

Replay attack: Timestamp in the PAPRSA is different current communication massage and present message.

Password file compromise: The square root modulo n (SQROOT) problem is to fine a for the given composite integer n and a^2 mod n. And RSA problem (problem for determine two primes from double two pries the given) is feasible transformation in polynomial time for SQROOT problem to same double two primes to n (polynomial time reducible) [15]. Therefore, RSA problem and SQROOT problems are computationally equivalent. Accordingly, attacker in PAPRSA is 1) safety for in case of only (y_1^2 mod N, y_2^2 mod N) compromise 2) safety for in case of supposing no eavesdropping only compromise of private key 3) safety for in case of supposing no eavesdropping only compromise of ((e, d), (p, q), N, y_1^2 mod N, y_2^2 mod N).

5.2 Efficient

The efficient of the PAPRSA is shown at Table 1.

Table 1. The efficient of the PAPRSA

Factors	Prover	Verifier
Pass	1	0
Random number generation	1	0
RSA encryption	2	0
RSA decryption	0	2
Modular multiplication	0	2

In PAPRSA, in case prover saves the verifier public key and in case prover no saves the verifier public key from application environment authentication system. In Fig. 2, the prover is show in case of no saves the verifier public key. It is 3-pass that the prover in case of no saves public key the verifier.

3-pass PAPRSA is described in Fig. 2.

PROTOCOL
Registering procedure
Input: id, P.
1. Get N=pq and (e, d);

2. Publish e and N;
3. Chooses y_1 in Z in random and then determine y_2 so that y_1-y_2=P;
4. Get $y_1^2 \bmod N$ and $y_2^2 \bmod N$, and then stores them at id.

Authentication procedure
Input: id, P.

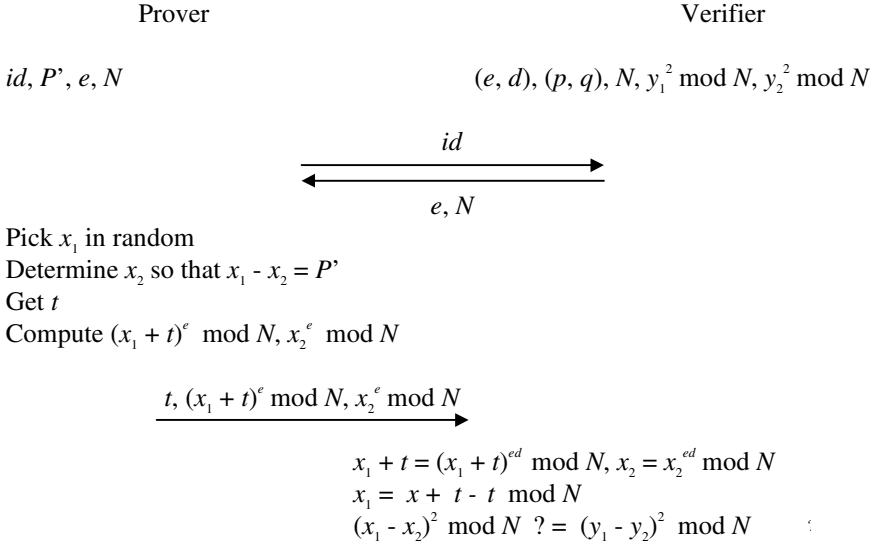


Fig. 2. 3-pass PAPRSA

5.3 Compared

The compared of on presented authentication protocol is shown at Table 2.
In password based authentication formally, Password file in Unix, OTP using human memorable password. But, It is not safe off-line dictionary attack and server compromise. The other side, in PAPRSA suggested safe to off-line dictionary attack and password file compromise.

Compared with challenge-response protocol, over twice of pass valve, it keeps safe to off-line dictionary attack. However, it needs addition hard to memory storing key, because it uses difficult key to remember. While PAPRSA is safe to off-line dictionary attack, because it uses possible password for people to remember, it doesn't need addition hard to memory.

Compared with zero-knowledge protocol, on twenty times pass value, it makes sever compromise file. But, it needs addition hard to memory storing key, because it uses difficult key for user to remember.

While PAPRSA is safe to password file compromise, it doesn't need addition hard to memory storing key by using possible password for people to remember.

Table 2. Compared authentication protocols

Formally	Protocols	Pass	Human-memorable	Attack		
				Replay	Pre-play	Dictionary
Password Based	Unix Salt	1	O	×[10]	× [19]	× [21]
	OTP	1	O	O [17]	× [22]	× [22]
Challenge response	Symmetric-key	> 2	×	O [13]	O [10]	O [10]
	Public-key	> 2	×	O [14]	O [10]	O [10]
Zero Knowl-edge		> 20	×	O [8]	O [11]	O [11]
Presented	PAPRSA	1	×	O	O	O
		3	O	O	O	O

5.4 Server Compromise

Attacker about server compromise basically uses secret information stored to server. About server compromise, it can be analyzed as following [19].

- For attacker in case of eavesdropping and no eavesdropping.
- For attacker in case of impersonation to user or to server.
- For attacker in case of using only password file compromise and server is private key.
- For attacker in case of doing off-line dictionary attack and no off-line dictionary attack.

However, even the zero knowledge proof protocols allow dictionary attacks and sever impersonation attacks if a sever file is compromised [23]. In PAPRSA is 1) safety for in case of only ($y_1^2 \bmod N$, $y_2^2 \bmod N$) compromise 2) safety for in case of supposing no eavesdropping only compromise of private key 3) safety for in case of supposing no eavesdropping only compromise of ((e, d), (p, q), N, $y_1^2 \bmod N$, $y_2^2 \bmod N$).

6 Conclusion and Future Study Subject

On information system, to login to special system, it uses authenticating protocol sending secret information safely between authenticating process and communication way confirming identification of user.

There are password based formally, challenge response protocol, zero- knowledge protocol on authenticating protocol suggested from now. Password based formally

uses possible password for people to memorize, but it is not safe to authenticating protocol attack like pre-play attack, off line dictionary attack, password file compromise. Challenge response protocol, zero- knowledge protocol need addition hard to memory storing key by using difficult key for user to memorize. Like this, from suggested authentication protocol, it is safe from attack like pre-play attack, off-line dictionary attack, password file compromise, because people use possible information to remember, we can figure out that it needs authenticating protocol that does not need addition hard to memory to store secret information.

On this paper, we define authenticating protocol scheme, PAP. PAP features that it expresses password choosing certain value in space expressing password, when any information was given to attacker, to determine which value is choose is selected on computationally in feasible. PAP considered safety about attack to analogize password from message eavesdropping, without analyzing safety of open, public key cryptosystem at the same time.

To encryption value expressing password basically with PAP, we suggested authenticating protocol, PAPRSA, using RSA. PAPRSA is powerfully secure from pre-play attack, off line dictionary attack, safe form compromise no eavesdropping. The pass value of PAPRSA is one, PAPRSA achieves twice with RSA encryption and decryption. And, it produces random once, achieves twice with modular multiplication.

After that, 1) From various compromise attack it designs safe authenticating protocol, 2) It designs identification protocol based PAP schema, 3) Protocol applied to help each other authenticating, session key public password expansion notion.

References

1. M. Bellare, D. Pointcheaval, and P. Rogaway, “Authenticated key exchange secure against dictionary attacks”, Advances in Cryptology Eurocrypt’00, LNCS Vol. 1807, Springer-Verlag, pp. 139-155, 2000.
2. S. M. Bellovin and M. Merrit, “Augmented encrypted key exchange: Password-based protocol secure against dictionary attack and password file compromise”, In ACM Security (CCS’93), pp. 244-250, 1993.
3. S. M. Bellovin and M. Merrit, “Encrypted key exchange: Password-based protocols secure against dictionary attack”, In Proceedings of IEEE Security and Privacy, pp. 72-84, 1992.
4. V. Boyko, P. MacKenzie, and S. Patal, “Provably secure password authenticated key exchange using Diffie-Hellman”, In B. Prenel, editor, Advances in Cryptology Eurocrypt’00, LNCS Vol. 1807, Springer-Verlag, pp. 156-171, 2000.
5. W. Diffie and H. E. Hellman, “New directions in cryptography”, IEEE Transactions on Information Theory, 22, pp. 644-654, 1976.
6. T. ElGamal, “A public-key cryptosystem and a signature scheme based on discrete logarithms,” IEEE Transactions on Information Theory, v. IT-31, n. 4, pp. 469-472, 1985.
7. U. Feige, A. Fiat and A. Shamir, “Zero knowledge proof of identity”, Journal of Cryptology, Vol. 1, pp. 77-94, 1983

8. Fiat and A. Shamir, “How to prove yourself: Practical solutions to identification and signature problems”, Advances in Cryptology-CRYPTO’ 86, LNCS 263, pp. 186-194, 1987.
9. L. Gong, “Optimal authentication protocols resistant to password guessing attacks”, In 8th IEEE Computer Security Foundations Workshop, pp. 24-29, 1995.
10. L. Gong, T. M. A. Lomas, R. M. Needham, and J. H. Saltzer, “Protecting poorly chosen secrets from guessing attacks”, IEEE Journal on Selected Areas in Communications, 11(5), pp. 648-656, June 1993.
11. L. C. Guillou and J. –J. Quisquater, “A practical zero-knowledge protocol to security microprocessor minimizing both transmission and memory”, Advances in Cryptology-EUROCRYPT ‘ 88, LNCS 330, pp. 123-128, 1988.
12. S. Halevi and H. Krawczyk, “Public-key cryptography and password protocols,” ACM Security (CCS’ 98), pp. 122-131.
13. ISO/IEC 9798-2, “Information technology-Security techniques-Entity authentication-Part 2: Mechanisms using symmetric encipherment algorithms”, International Organization for Standardization, Geneva, Switzerland, 1994.
14. ISO/IEC 9798-4, “Information technology-Security techniques-Entity authentication-Part 4: Mechanisms using a cryptographic check function”, International Organization for Standardization, Geneva, Switzerland, 1995.
15. D. Jablon, “Strong password-only authenticated key exchange”, ACM Computer Communication Review, ACM SIGCOMM, Vol. 26, No. 5, pp. 5-20, October 1996.
16. N. Koblitz, “Eliptic curve cryptosystems,” Mathematics of Computation, v. 48, n. 177, pp. 203-209, 1987.
17. L. Lamport, "Password authentication with insecure communication", Communications of the ACM, Vol. 24, pp. 770-772, 1981.
18. R. J. McEliece, “A public key cryptosystem based on algebraic coding theory,” Deep Space Network Progress Report 42-44, Jet Propulsion Laboratory, California Institute of Technology, pp. 42-44, 1978.
19. J. Menezes, P. C. van Oorschot and S. A. Vanstone, Applied Cryptography, CRC press, 1997.
20. R. C. Merkle, Secrecy, Authentication, and Public Key Systems, UMI Research Press, Ann Arbor, Michigan, 1979.
21. R. Morris and K. Thompson, "Password security: a case history", Communications of the ACM, Vol. 22, pp. 594-597, 1979.
22. Chris J. Mitchell and Liqun Chen, “Comment on the S/KEY user authentication scheme”,(ASPect)
23. T. Kwon, “Authentication and key agreement via memorable password,”2000, available from http://eprint.iacr.org/2000/026

Effective Packet Marking Approach to Defend against DDoS Attack ·

Heeran Lim and Manpyo Hong

Internet Immune System Laboratory, Ajou University, Wonchun-dong, Youngtong-gu, Suwon, Kyounki -do, South Korea
lhr5456@empal.com, mphong@ajou.ac.kr

Abstract. Distributed Denial of Service (DDoS) attack is one of the remained problems that has still not solved. In most case, these attacks raise the flows of packets with spoofed IP, thus it is too hard to decide what packet is attack packet or not. Up to now, various studies have been proposed to defend against DDoS attacks. We don't still have had a fitting solution to solve it. Abraham et al. presented Pi that is scheme to marks packet's path.Pi is a scheme that marks traveling information of each packet on itself. Pi is a new, simple and robust approach but, Pi can have poor marking value filled with garbage when there are too small routers which participated in marking. This brings the loss of information. We propose a new marking approach that improves the previous Pi marking scheme. It has a higher accuracy value than previous Pi making.

1 Instruction

Distributed Denial of Service (DDoS) attack is the remained problem that has still not solved. In most case of DDoS attack, attacker has lots of compromised hosts and uses them to send large number of packets with automatic attacking tools (such as Tribal Flooding Network (TFN), TFN2K, Trinoo, and Stacheldraht) to a single victim server. For example, in October 21, 2002, DDoS attack was launched against 13 root servers which are used by the other Internet's Domai0n Name Servers (DNS) [1]. In January 25, 2003, a worm called Slammer [2] infected more than 90 percents of vulnerable hosts within 10 minutes and it cause explosive traffics on the whole internet. This attack is targeted to anonymous hosts because of its spreading strategy – randomly selecting.

It is too difficult to differentiate between the attack packet and normal packets. That is a main problem of defense against DDoS attack. There have been many studies to defend against DDoS attack [3, 4, 5, 6, 7, 8, 9, 10, 11, 12]. Some previous approach investigated inside the packets to confirm if the given packets are malicious or not. The other has considered the topologies of attack traffic. These approaches needs too much time to defend against DDoS attack. Abraham Yaar et al. introduced a packet

∗ This work was supported by grant No. (R05-2003-000-11235-0) from the Basic Research Program of the Korea Science & Engineering Foundation.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 708–716, 2004.
© Springer-Verlag Berlin Heidelberg 2004

marking scheme, called Pi (Path Identification) to defend against this attack [13, 14]. Pi is a marking scheme that records traverse information on each packet. Pi is a simple and robust but, it can have a little poor marking value filled with garbage when there are too small routers which participates in marking. We used trace-routed maps of real Internet topologies (e.g., CAIDA's Skitter Map[15]) to compare with Pi and to evaluate our approach.
In this paper, we present a new marking scheme that improves previous Pi marking. We introduce new approach of Pi marking scheme in the section 3, and present effectiveness of the new marking scheme through the experiment in the section 5.

2 Overview of Pi

Pi is packet marking scheme that leaves traverse information of packet in each packet. It uses IP address of the routers as a part of traverse information. There are two types of routes. One is legacy router which does not participate in marking, and the other is Pi router which participates in marking. Pi router marks the last n bits of IP address of router on the marking field of packet. Pi uses last 1 bit or 2 bits of IP addresses or router as traverse information and, we call it 1-bit Pi and 2-bit Pi each other. Pi can also use the last n bits from the hash of their IP addresses for more uniform distribution of the last bits of the IP addresses of the routers. Pi uses TTL value of IP packet to decide the location within the marking field. The marking field is separated into 16/n marking sections and uses the TTL modulo 16/n to decided a marking section in the marking field. As a result Pi has the path information of each packet. Thus, if we know the Pi values of packets that attackers send, we can filter out the malicious packets on time. Pi can conduct both packet marking on the router and packet filtering on the victims within extremely short time.

3 Motivation

Original Pi marking uses TTL value of packet to decide the index of marking section, and the value is TTL modulo [16/n]. It causes problems as follow.
It can make some sections unmarked when the legacy routers exist. Legacy router may decrease the TTL value of packet and remains the garbage value in some sections. This is vulnerable point that can be abused by attacker. Consider that an attacker fabricates initial value of marking field of packet. The unmarked section in marking field will be filled with garbage value which attacker fabricates. Thus, the marking value of Pi will be variable and far from intended Pi value. It would raise false positive.
Above problem may be solved if there are too many routers on the path of packets. We can consider the problem into two cases. In the first case of 1-bit marking Pi, it needs at least 16 hops on path to fill whole marking sections. But, only 20% paths have 16 or more hops. Because there may be many legacy routers, it is too difficult to mark on all sections. In the other case of 2-bit marking, it needs at least 8 hops – 16/2

- to fill out all marking sections. The paths with 8 or more hops reach 88%. But if there are too many legacy routers, this is difficult too.
For solving above problem, we propose a new approach of Pi marking scheme not using TTL. When Pi router gets a packet, it shifts Pi value to the left direction. Then it marks its own marking value on the most right section of marking field regardless of TTL value. In case of 2-bit marking, proposed new Pi can have marking values in whole sections if there are only 8 Pi routers on the path. But previous Pi may not mark in all sections because of characteristic of overwriting Pi value in marked section again.

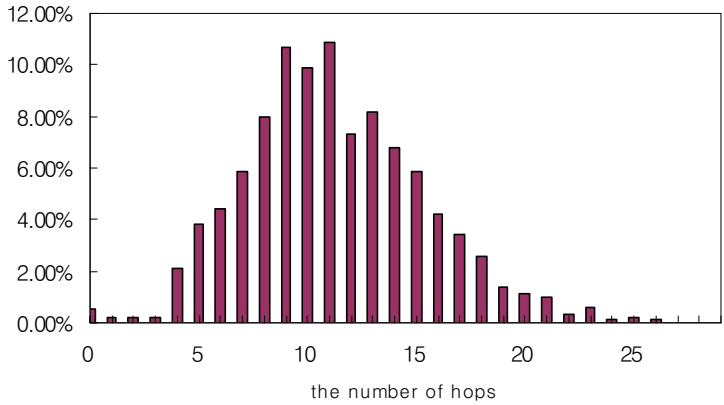


Fig. 1. The distribution of hop counts

4 New Approach of Pi Marking

Proposed new marking scheme also separates the marking field into [16/n] marking sections and uses the last n bits from pure or hashed IP address of the routers. The method of deciding a marking section is different from the previous Pi.

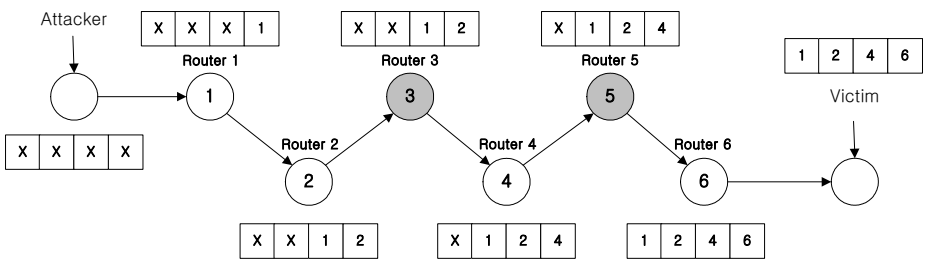


Fig. 2. Marking scenario of proposed Pi

Pi router always marks in the most right section of marking field after shifting n-bit to the left. Figure 2 and 3 depicts proposed new Pi and previous Pi. We used 4 sections of marking field for simplicity. Shaded circles (router3 and router5) are legacy routers which are not participated in marking. The other circles present Pi routers which would do marking.

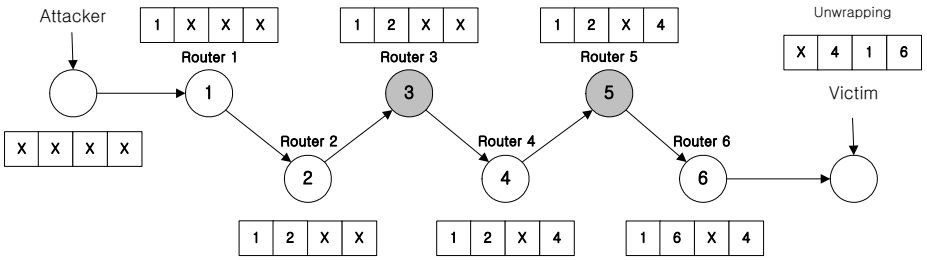


Fig. 3. Marking scenario of previous Pi

In Figure 2 and 3, Proposed Pi does not have garbage value in marking field but, previous Pi marking has garbage value in the first section. The previous Pi marking scheme can garbage value according to the topology of routers.

The fundamental goal of our approach is that reduce unmarked sections in the marking field as possible. Our approach is marking more than previous Pi marking. When there are seven Pi routers on the path of packet, new Pi marking has only one garbage section but previous Pi may have one or more garbage sections according to hop counts.

5 Experiment

In this section, we evaluate that proposed marking scheme is more effective than previous Pi. We used real Internet topology that is CAIDA's Skitter MAP[3]. This data were created by one host that sends traceroute information to randomly chosen destinations. This consists of lots of path which is from one host to lots of destination. In our experiment, we used the n = 2 bit scheme because it is difficult to identify routers with only 1 bit. Moreover, Pi values can have garbage value easily because there are lots of paths under 16 hops. In the case of n = 3 bit scheme, it can mark only 5 routers (namely, only 15 bit would be used), it would limit our marking space to 215 = 32,768. If use n = 4 bit scheme, it would only have 4 router information. Moreover the routers beside victim are same or similar. 2 bit scheme router has 8 routers information and 65,535 marking space.

We have had two experiments. One is to confirm the distribution of Pi values, and the other has been done to find out the correlation between the number of cases of Pi value and the number of paths.

5.1 The Distribution Pi Value

Figure 4 and 5 shows distribution of Pi values of both original Pi and proposed new Pi. We extracted Pi values from 785,000 paths to the one host in the traceroute map. Each uses the random value to initialize value in marking field, IP identification field of packet.

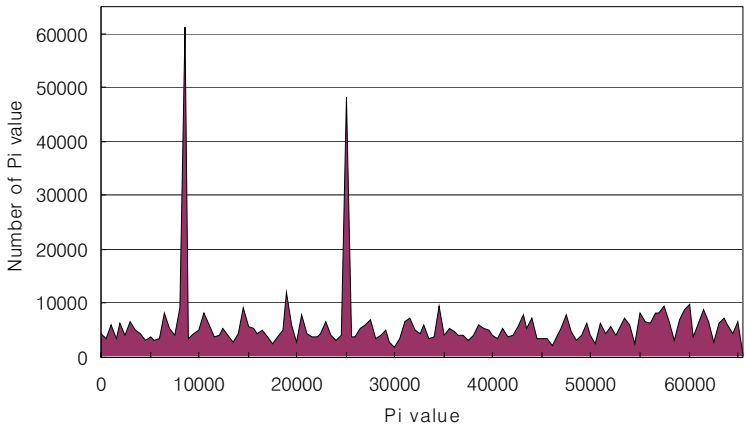


Fig. 4. Original Pi scheme, Legacy 25%, The distribution of Pi value

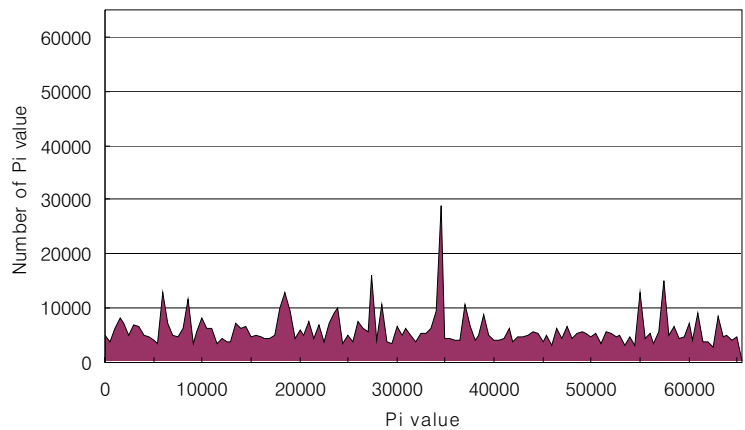


Fig. 5. New Pi scheme, Legacy 25%, The distribution of Pi value

The horizontal axis indicates Pi values and the vertical axis indicates the number of each Pi value.

In see Figure 4.depicts the result of distribution of original Pi value with 25% legacy router. It seems to be biased to the value near section of 8,000 and 25,000.

Biased value can cause false positive or false negative. If biased value is Pi value of attack packets, we may filter out normal packets too. As a result, legitimate users can not use a service any more. To solve this situation, we need to make Pi distributed uniformly. The new Pi also has a little biased to some value, but it does more little than the previous Pi. The new Pi has more uniformly distributed form. So it would reduce the false positive.

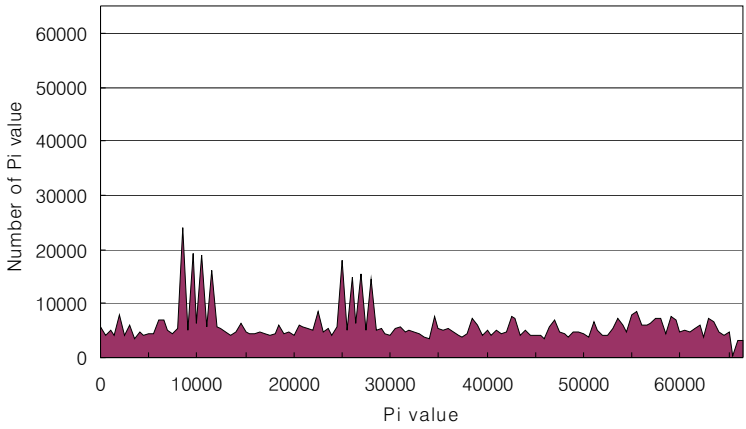


Fig. 6. Original Pi scheme, Legacy 50%, The distribution of Pi value

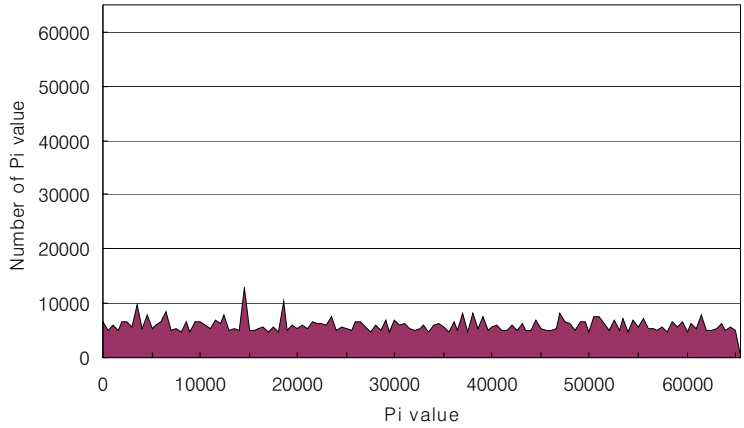


Fig. 7. New Pi scheme, Legacy 50%, The distribution of Pi value

In the case of using the 50% legacy router, two schemes have more uniformly distributed result than the result of experiment using the 25% legacy routers. The

more there are many routers, the distribution will be better. The new Pi has also better distribution than the previous Pi as figure 6 and 7.

5.2 The Number of Cases According to the Number of Paths

In the second experiments, we extract Pi values from one path 10 times with randomly generated marking field. There were 1,000 paths.

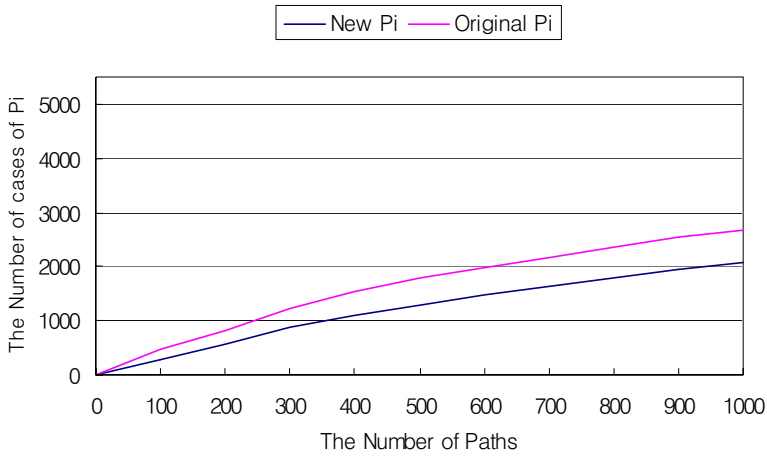


Fig. 8. the number of cases of Pi, Legacy 25%

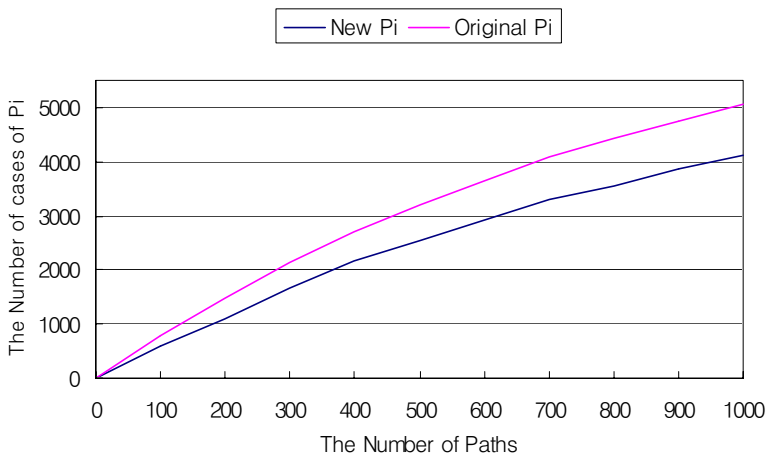


Fig. 9. the number of cases of Pi, Legacy 50%

We can see that the number of cases of Pi values of New Pi is smaller than the previous Pi. Let N is the number of cases of paths, the scope of N is K < N <= K * 10. Thus the number of cases of Pi is between 500 and 500*10 in case of 500 paths.

Figure 8 and Figure 9 are the result of comparing the number of cases of the original Pi and new Pi according to the paths. We can find that new Pi has smaller number of Pi values than previous Pi. Maybe it is caused by the number of unmarked sections. In the case of new Pi, the number of unmarked sections is smaller than previous Pi. Both original Pi and new Pi get the Pi value from the same paths. But, they have different result. The reason is that the number of Pi routers is same but location of marking section is different. The previous Pi can mark on the same marking section again because it selects the location of marking section with TTL. But, new Pi does not mark on the same marking sections if there are 8 or less Pi routers.

6 Conclusions

We presented a new Pi marking scheme that improves previous Pi marking. The purpose of the new Pi is minimized the garbage value in marking field. The method of deciding a marking section is different from the previous Pi. Our experiments showed that new Pi scheme makes more uniform distribution of Pi value than previous Pi and it is better scheme than previous Pi. Thus there will be less false positive. The proposed scheme will be a good approach to defend against DDoS attack.

References

1. Ryan Naraine, Massive. DDoS Attack Hit DNS Root Servers, eSecurityPlanet.com (Oct 2002) http://www.esecurityplanet.com/trends/article.php/10751_1486981
2. Inside the Slammer Worm, http://www.computer.org/security/v1n4/j4wea.htm
3. Denial of Service Attacks, CERT (1997)
4. XiaoFeg Wang, Michael K. Reiter. Defending Against Denial-of-Service Attacks with Puzzle Auctions, In Proceedings of the 2003 Security and Privacy Symposium (May. 2003)
5. J. Ioannidis and S. M. Bellovin. Implementing Pushback: Router-based defense against DDoS attacks, In Proceedings of the Symposium on Network and Distributed Systems Security(NDSS 2002) (feb 2002).
6. Sterne, D.; Djahandari, K.; Balupari, R.; La Cholter, W.; Babson, B.; Wilson, B.; Narasimhan, P.; Purtell, A.; Schnackenberg, D.; Linden, S. Active network based DDoS defense, 193- 203
7. Luiz Gustavo, Martins Arruda. Around Network Intrusion Prevention Systems
8. Kihong Park, Heejo Lee. On the Effectiveness of RouteBased Packet Filtering for Distributed DoS Attack Prevention in PowerLaw Internets SIGCOMM'01 (2001) San Diego, California, USA.
9. Kashiwa, D.; Chen, E.Y.; Fuji, H. Active shaping: a countermeasure against DDoS attacks, (ECUMN 2002) 2nd European Conference on, 8-10 (April 2002) 171 - 179
10. Jelena Mirkovic, Peter Reiher. A Taxonomy of DDoS Attack and DDoS Defense Mechanisms

11. Tao Peng, Christopher Leckie, Kotagiri Ramamohanarao. Protection from Distributed Denial of Service Attack Using History-based IP Filtering
12. Jun Xu, Wooyong Lee. Sustaining Availability of Web Services under Distributed Denial of Service Attacks
13. A. Perrig, D. song, and A. Yaar. Pi: A Path Identification Mechanism to Defend against DDoS Attacks, In Proceedings of the 2003 Security and Privacy Symposium (May. 2003)
14. A. Perrig, D. song, and A. Yaar. Pi: A new defense mechanism against IP spoofing and DDoS attacks, Technical Report CMU-CS-02-207, Carnegie Mellon University, School of computer Science (Dec. 2002)
15. Caida. Skitter., http://www.caida.org/tools/measurement/skitter/ (2000)

A Relationship between Security Engineering and Security Evaluation

Tai-hoon Kim[1] and Haeng-kon Kim[2]

[1]KISA, 78, Garak-Dong, Songpa-Gu, Seoul, Korea
{taihoon, shkim}@kisa.or.kr
http://www.kisa.or.kr
[2]Catholic University of Daegu , 330, Hayangup, Kyungsan, Kyungbuk, Korea
{hangkon}@cu.ac.kr
http://selab.cataegu.ac.kr/index.html

Abstract. The Common Criteria (CC) philosophy is to provide assurance based upon an evaluation of the IT product or system that is to be trusted. Evaluation has been the traditional means of providing assurance. It is essential that not only the customer's requirements for software functionality should be satisfied but also the security requirements imposed on the software development should be effectively analyzed and implemented in contributing to the security objectives of customer's requirements. Unless suitable requirements are established at the start of the software development process, the resulting end product, however well engineered, may not meet the objectives of its anticipated consumers. By the security evaluation, customer can sure about the quality of the products or systems they will buy and operate. In this paper, we propose a selection guide for IT products by showing relationship between security engineering and security evaluation and make help user and customer select appropriate products or systems.

1 Introduction

The CC, the International Organization for Standard (IS) 15408 [1-3], is a standard for specifying and evaluating the security features of IT products and systems, and is intended to replace previous security criteria such as the TCSEC. This evaluation process establishes a level of confidence that the security functions of such products and systems, and the assurance measures applied to them, must meet.

With the increasing reliance of society on information, the protection of that information and the systems contain that information is becoming important. In fact, many products, systems, and services are needed and used to protect information. The focus of security engineering has expanded from one primarily concerned with safeguarding classified government data to broader applications including financial transactions, contractual agreements, personal information, and the Internet. These trends have elevated the importance of security engineering [4].

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 717–724, 2004.
© Springer-Verlag Berlin Heidelberg 2004

When we are making some kinds of software products, the customer's requirements must be implemented to software perfectly, but this is not sufficient. The secure software may be implemented by not only applying Firewall or IDS but also considering security requirement appended to customer's requirement. In this paper, we will propose a concept of security requirements appended to customer's requirements and show the relationship between security requirement and the security evaluation of implementation.

2 A Review of Common Criteria

The multipart standard ISO/IEC 15408 defines criteria, which for historical and continuity purposes are referred to herein as the Common Criteria (CC), to be used as the basis for evaluation of security properties of IT products and systems. By establishing such a common criteria base, the results of an IT security evaluation will be meaningful to a wider audience.

The CC will permit comparability between the results of independent security evaluations. It does so by providing a common set of requirements for the security functions of IT products and systems and for assurance measures applied to them during a security evaluation. The evaluation process establishes a level of confidence that the security functions of such products and systems and the assurance measures applied to them meet these requirements. The evaluation results may help consumers to determine whether the IT product or system is secure enough for their intended application and whether the security risks implicit in its use are tolerable.

The CC is presented as a set of distinct but related parts as identified below.

Part 1, Introduction and general model, is the introduction to the CC. It defines general concepts and principles of IT security evaluation and presents a general model of evaluation. Part 1 also presents constructs for expressing IT security objectives, for selecting and defining IT security requirements, and for writing high-level specifications for products and systems. In addition, the usefulness of each part of the CC is described in terms of each of the target audiences.

Part 2, Security functional requirements, establishes a set of functional components as a standard way of expressing the functional requirements for TOEs (Target of Evaluations). Part 2 catalogues the set of functional components, families, and classes.

Part 3, Security assurance requirements, establishes a set of assurance components as a standard way of expressing the assurance requirements for TOEs. Part 3 catalogues the set of assurance components, families and classes. Part 3 also defines evaluation criteria for PPs (Protection Profiles) and STs (Security Targets) and presents evaluation assurance levels that define the predefined CC scale for rating assurance for TOEs, which is called the Evaluation Assurance Levels (EALs).

In support of the three parts of the CC listed above, it is anticipated that other types of documents will be published, including technical rationale material and guidance documents.

3 Protection Profile for Specification of Security Requirements

A PP defines an implementation-independent set of IT security requirements for a category of TOEs. Such TOEs are intended to meet common consumer needs for IT security. Consumers can therefore construct or cite a PP to express their IT security needs without reference to any specific TOE.

The purpose of a PP is to state a security problem rigorously for a given collection of systems or products (known as the TOE) and to specify security requirements to address that problem without dictating how these requirements will be implemented. For this reason, a PP is said to provide an implementation-independent security description. A PP thus includes several related kinds of security information (See the Fig. 1).

A description of the TOE security environment which refines the statement of need with respect to the intended environment of use, producing the threats to be countered and the organizational security policies to be met in light of specific assumptions.

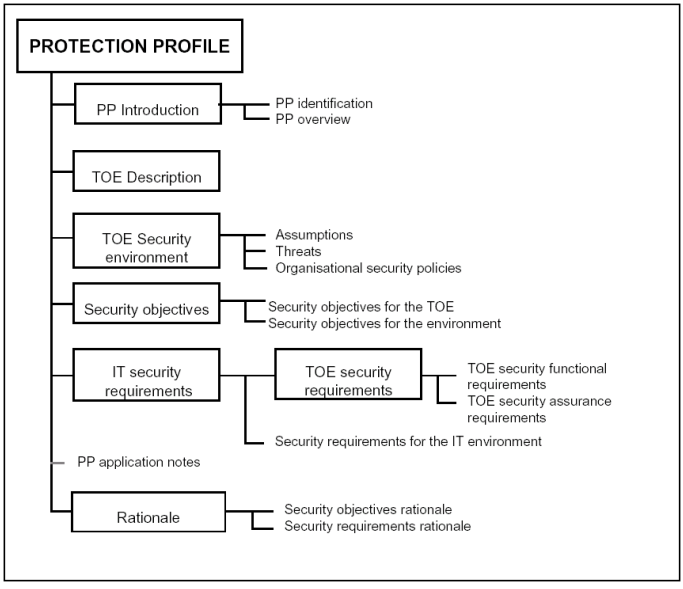


Fig. 1. Protection Profile content

4 Security Engineering

As mentioned earlier, the security engineering is focused on the security requirements for implementing security in software or related systems.

In fact, the scope of security engineering is very wide and encompasses:

•the security engineering activities for a secure software or a trusted system addressing the complete lifecycle of: concept definition, analysis of customer's requirements, high level design and low level design, development, integration, installation and generation, operation, maintenance end de-commissioning;
•requirements for product developers, secure systems developers and integrators, organizations that develop software and provide computer security services and computer security engineering;
•applies to all types and sizes of security engineering organizations from commercial to government and the academe.

The security engineering should not be practiced in isolation from other engineering disciplines. Maybe the security engineering promotes such integration, taking the view that security is pervasive across all engineering disciplines (e.g., systems, software and hardware) and defining components of the model to address such concerns.

The main interest of customers and suppliers may be not improvement of the development of security characteristics but performance and functionality. If developers consider some security-related aspects of software developed, maybe the price or fee of software more expensive. But if they think about that a security hole can compromise whole system, some cost-up will be appropriate.

The field of security engineering has several generally accepted principles, but it currently lacks a comprehensive framework for evaluating security engineering practices. The ISO/IEC 21827 (SSE-CMM), by identifying such a framework, provides a way to measure and improve performance in the application of security engineering principles. It must be stressed that security engineering is a unique discipline, requiring unique knowledge, skills, and processes which warrants the development of a distinct CMM for security engineering.

This does not conflict with the premise that security engineering is done in context with systems engineering. In fact, having well-defined and accepted systems engineering activities will allow security engineering to be practiced effectively in all contexts.

5 Relationship between Security Engineering and Evaluation

The main objective of application of security engineering is to provide assurance about the software or system to customer. The assurance level of a software or system may be the critical factor has influence on deciding purchase.

Traditionally, assurance has only been associated with computer hardware and software; however, there are assurance aspects that apply to the delivery of security services, such as a threat and risk assessment (TRA). In general, IT security assurance is the confidence one has that the Target of Evaluation (TOE) computer system will perform as designed. From a security standpoint, this translates into confidence that the IT product or system enforces its security policy. ISO 15408 Part 1 defines assurance as: *Grounds for confidence that an entity meets its security objectives.*

The above assurance definition, although correct, is limited to products and systems, which does not account for the delivery of services (i.e. managed firewall security services). Therefore, to be more generic and accommodate security services, the assurance definition is redefined as: *Grounds for confidence that a deliverable meets its security objectives.*

This definition has been reworded to be generic to support security products, systems, and services while still being in line with the ISO 15408 assurance definition. Both products and services are delivered and both have measurable security attributes that can be verified to meet their security objectives. For example, a product meets its security policy and a service organization delivering a service satisfies the security objectives related to the service delivery. The security objectives for a service could be a successful background checks of the consultant delivering the service or having the service delivered in a predictable and repeatable manner.

Following the discussion of assurance, it can be seen that an assurance approach can be used to instill confidence that any deliverable (IT component or security service) will function as claimed. Similar to an IT development lifecycle, an IT security service is also developed and delivered and therefore an assurance approach can be used to measure the assurance of the IT component or service. In the security field, assurance of security services have not been addressed sufficiently due to the main focus on IT security products and systems requiring most of the available resources. Furthermore, assurance of IT components is more tangible than services. This technical report includes security services in the security assurance framework.

An assurance method can be used in two ways. An assurance method can be used to measure assurance or to introduce assurance into an IT component or security service deliverable. The same assurance model and guidance can be used by organizations as assurance goals to improve their deliverable. Therefore, these methods can be applied to any phase of an IT development lifecycle depending on the type of assurance desired. Similarly, an IT security service is also developed and delivered according to a lifecycle model and therefore the assurance obtained is a direct result of the assurance method used.

6 Application of Security Engineering for Evaluation

A wide variety of organizations can apply security engineering to their work such as the development of computer programs, software and middleware of applications programs or the security policy of organizations. Appropriate approaches or methods and practices are therefore required by product developers, service providers, system

integrators, system administrators, and even security specialists. Some of these organizations deal with high-level issues (e.g., ones dealing with operational use or system architecture), others focus on low-level issues (e.g., mechanism selection or design), and some do both.

The security engineering may be applied to all kinds of organizations. Use of the security engineering principle should not imply that one focus is better than another is or that any of these uses are required. An organization's business focus need not be biased by use of the security engineering.

Based on the focus of the organization, some, but not all, of approaches or methods of security engineering may be applied very well. In fact, generally, it is true that some of approaches or methods of security engineering can be applied to increase assurance level of software.

There are many methodologies for software development, and security engineering does not mandate any specific development methodology or life cycle model. In this paper, we used the methodology of waterfall.

6.1 Append Security-Related Requirements

For the development of software, the first objective is the perfect implementation of customer's requirements. And this work may be done by very simple processes. However, if the software developed has some critical security holes, the whole network or systems that software installed and generated are very vulnerable.

Therefore, developers or analyzers must consider some security-related factors and append a few security-related requirements to the customer's requirements. Fig.2 depicts the idea about this concept.

The processes based on the refinement of the security-related requirements are considered with the processes of software implementation.

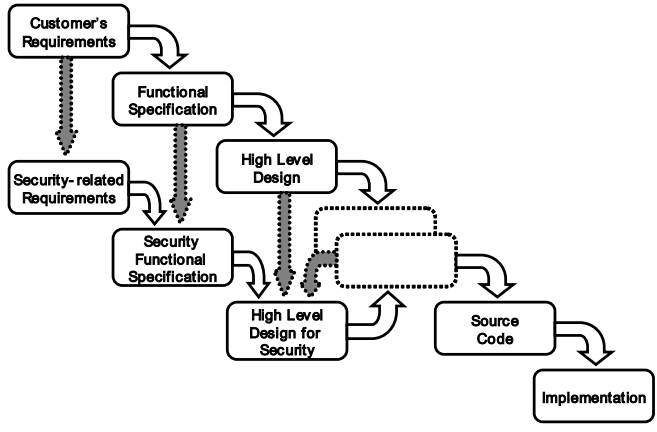


Fig. 2. Append security-related requirements

6.2 Implementation of Appended Security-Related Requirements

Developers can reference the ISO/IEC 15408, Common Criteria (CC), to implement security-related requirements appended.

The multipart standard ISO/IEC 15408 defines criteria, which for historical and continuity purposes are referred to herein as the CC, to be used as the basis for evaluation of security properties of IT products and systems. By establishing such a common criteria base, the results of an IT security evaluation will be meaningful to a wider audience.

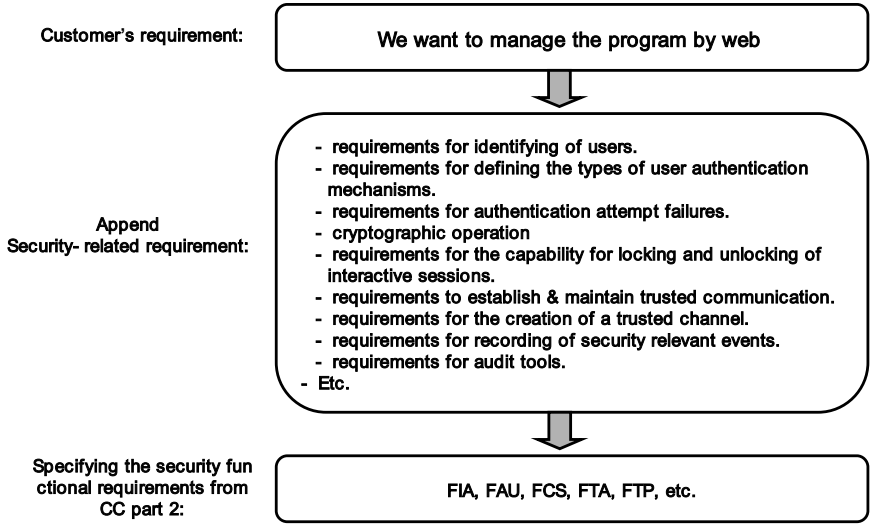


Fig. 3. Specifying the security functional requirements

The CC will permit comparability between the results of independent security evaluations. It does so by providing a common set of requirements for the security functions of IT products and systems and for assurance measures applied to them during a security evaluation. The evaluation process establishes a level of confidence that the security functions of such products and systems and the assurance measures applied to them meet these requirements. The evaluation results may help consumers to determine whether the IT product or system is secure enough for their intended application and whether the security risks implicit in its use are tolerable.

For example, if the customer's requirement 'management by web' is specified, we can append some security-related requirements (See the Fig.3) and specified that requirements form CC part 2:

(1) Customer's requirement: We want to manage the program by web
(2) Appended requirements: Next security-related requirements are appended by developers or designers
(3) Specifying the security functional requirements from CC part 2

7 Conclusions

When we are making some kinds of software products, the customer's requirements must be implemented to software perfectly, but this is not sufficient. The secure software may be implemented by not only applying Firewall or IDS but also considering security requirement appended to customer's requirement.

Customers or users of these softwares want to assure the security and therefore rely on the evaluation. In this paper, we will propose a concept of security requirements appended to customer's requirements and show the relationship between security requirement and the security evaluation of implementation.

For the development of software, the first objective is the perfect implementation of customer's requirements. However, if the software developed has some critical security holes, the whole network or systems that software installed and generated may be very vulnerable.

Therefore, developers or analyzers must consider some security-related factors and append a few security-related requirements to the customer's requirements. The processes based on the refinement of the security-related requirements are considered with the processes of software implementation.

References

1. ISO. ISO/IEC 15408-1:1999 Information technology - Security techniques - Evaluation criteria for IT security - Part 1: Introduction and general model
2. ISO. ISO/IEC 15408-2:1999 Information technology - Security techniques - Evaluation criteria for IT security - Part 2: Security functional requirements
3. ISO. ISO/IEC 15408-3:1999 Information technology - Security techniques - Evaluation criteria for IT security - Part 3: Security assurance requirements
4. ISO. ISO/IEC 21827 Information technology – Systems Security Engineering Capability Maturity Model (SSE-CMM)
5. Tai-Hoon Kim, Byung-Gyu No, Dong-chun Lee: Threat Description for the PP by Using the Concept of the Assets Protected by TOE, ICCS 2003, LNCS 2660, Part 4, pp. 605-613

A Relationship of Configuration Management Requirements between KISEC and ISO/IEC 15408

Hae-ki Lee[1], Jae-sun Shim[2], Seung Lee[3], and Jong-bu Kim[4]

[1]Chung Cheong Univ., 330, Wallgok-Ri, Gnagnae-Myoun, Cheongwon-Gun, Chungbuk-Do
lhk77@ok.ac.kr
http://www.ok.ac.kr/english/sub_1.jsp

[2]Samcheok National Univ., San 253, Gyo-Dong, Samcheok-si, Gangwon-do, Korea
myongo@samcheok.ac.kr
http://www.samcheok.ac.kr/

[3]Daelim Colleage, 526-7, Bisan-Dong, Dongan-Gu, Anyang-city, Gyeonggi-Do
slee@daelim.ac.kr
http://www.daelim.ac.kr

[4]Induk Univ., San 75, Wallgye-Dong, Nowon-Gu, Seoul, Korea
jbkim@induk.ac.kr
http://www.induk.ac.kr

Abstract. There are very many assurance methods and most of them are listed in the ISO/IEC 15443. The objective of ISO/IEC 15443 is to present a variety of assurance methods, and to guide the IT Security Professional in the selection of an appropriate assurance method (or combination of methods) to achieve confidence that a given IT security product, system, service, process or environmental factor satisfies its stated security assurance requirements. In the Part 2 of ISO/IEC 15443, many assurance methods and approaches proposed by various types of organizations are introduced, and in the Part 3 of ISO/IEC 15443, Analysis of Assurance Methods, various assurance methods are analyzed with respect to relationships and equivalency, effectiveness and required resources. This analysis may form the basis for determining assurance approaches and making trade-offs among the various factors for given security applications. The materials in Part 3 contain the mapping SSE-CMM (System Security Engineering Capability Maturity Model) to CC (Common Criteria), T-CMM (Technical-CMM) to CC and so forth. SSE-CMM and T-CMM are listed in the Part 2, and the Part 3 selects the assurance method or approach in the Part 2 to verify the relationship of them by using the concepts based on the software engineering. An assurance method KISEC (Korea Information Security Evaluation Criteria) will be included in ISO/IEC 15443 Part 2, and the research about the relationship with respect to software engineering between KISEC and CC is needed. This paper is about the research for the relationship of assurance requirements for configuration management between KISEC and CC. This paper will help the development company of IT product and system to understand the evaluation criteria CC and prepare for an evaluation.

Keywords: Assurance requirement, configuration management, common criteria, KISEC

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 725–734, 2004.
© Springer-Verlag Berlin Heidelberg 2004

1 Introduction

The CC is a standard for specifying and evaluating the security features of IT products and systems, and is intended to replace previous security criteria such as the TCSEC. ISO/IEC 15408, The Common Criteria (CC), philosophy is to provide assurance based upon an evaluation of the IT product or system that is to be trusted [1-3], and an evaluation has been the traditional means of providing assurance.

In fact, there are many evaluation criteria in the world and the most of these criteria may be included in ISO/IEC 15443 Part 2 [4-6]. TCSEC (The Trusted Computer System Evaluation Criteria) used in the USA, ITSEC (The European Information Technology Security Evaluation Criteria) used in the Europe, and CTCPEC (The Canadian Trusted Computer Product Evaluation Criteria) used in the Canada existed, and they have evolved into a single evaluation entity, the CC.

For the achievement of the objective, ISO/IEC 15443 lists and examines assurance methods and approaches proposed by various types of organizations in the Part 2, and analyses the various assurance methods with respect to relationships in the Part 3. This analysis may form the basis for determining assurance approaches and making trade-offs among the various factors for given security applications. The material in Part 3 targets the IT security professional who must select assurance methods and approaches.

KISEC will be included in ISO/IEC 15443 Part 2, and the research about the relationship between KISEC and CC is processing. This paper is a part of research about the relationship for assurance requirements between KISEC and CC, and deals only configuration management. This result may be included in ISO/IEC 15443 Part 3, and will help the development company of IT product or system to understand the evaluation criteria CC and KISEC and to prepare an evaluation.

2 Overview

2.1 Overview of ISO/IEC 15443

As mentioned earlier, the objective of ISO/IEC 15443 is to present a variety of assurance methods, and to guide the IT Security Professional in the selection of an appropriate assurance method to achieve confidence.

In pursuit of this objective, ISO/IEC 15443 comprises the following:

- a framework model to position existing assurance methods and to show their relationships;
- a collection of assurance methods, their description and reference;
- a collection of assurance elements which may be part of such methods or which may individually contribute to assurance;
- a presentation of common and unique properties specific to assurance methods and elements;

- qualitative, and where possible, quantitative comparison of existing assurance methods and elements;
- identification of assurance schemes currently associated with assurance methods;
- a description of relationships between the different assurance methods and elements; and
- a guidance to the application, composition and recognition of assurance methods.

ISO/IEC 15443 is organized in three parts to address the assurance approach, analysis, and relationships as follows:

Part 1, Overview and Framework, provides an overview of the fundamental concepts and general description of the assurance methods and elements. This material is aimed at understanding Part 2 and Part 3 of ISO/IEC 15443. Part 1 targets IT security managers and others responsible for developing a security assurance program, determining the assurance of their deliverable, entering an assurance assessment audit (e.g. ISO 9000, SSE-CMM, ISO/IEC 15408), or other assurance activities.
Part 2, Assurance Methods, describes a variety of assurance methods and approaches and relates them to the assurance framework model of Part 1. The emphasis is to identify qualitative properties of the assurance methods and elements that contribute to assurance, and where possible, to define assurance ratings. This material is catering to an IT security professional for the understanding of how to obtain assurance in a given life cycle stage of product or service.
Part 3, Analysis of Assurance Methods, analyses the various assurance methods with respect to relationships and equivalency, effectiveness and required resources. This analysis may form the basis for determining assurance approaches and making trade-offs among the various factors for given security applications. The material in this part targets the IT security professional who must select assurance methods and approaches.

2.2 Overview of Common Criteria

The multipart standard ISO/IEC 15408 defines criteria, which for historical and continuity purposes are referred to herein as the Common Criteria (CC), to be used as the basis for evaluation of security properties of IT products and systems. By establishing such a common criteria base, the results of an IT security evaluation will be meaningful to a wider audience.

The CC will permit comparability between the results of independent security evaluations. It does so by providing a common set of requirements for the security functions of IT products and systems and for assurance measures applied to them during a security evaluation. The evaluation process establishes a level of confidence that the security functions of such products and systems and the assurance measures applied to them meet these requirements. The evaluation results may help consumers

to determine whether the IT product or system is secure enough for their intended application and whether the security risks implicit in its use are tolerable.

The CC is presented as a set of distinct but related parts as identified below.

Part 1, Introduction and general model, is the introduction to the CC. It defines general concepts and principles of IT security evaluation and presents a general model of evaluation. Part 1 also presents constructs for expressing IT security objectives, for selecting and defining IT security requirements, and for writing high-level specifications for products and systems. In addition, the usefulness of each part of the CC is described in terms of each of the target audiences.

Part 2, Security functional requirements, establishes a set of functional components as a standard way of expressing the functional requirements for TOEs (Target of Evaluations). Part 2 catalogues the set of functional components, families, and classes.

Part 3, Security assurance requirements, establishes a set of assurance components as a standard way of expressing the assurance requirements for TOEs. Part 3 catalogues the set of assurance components, families and classes. Part 3 also defines evaluation criteria for PPs (Protection Profiles) and STs (Security Targets) and presents evaluation assurance levels that define the predefined CC scale for rating assurance for TOEs, which is called the Evaluation Assurance Levels (EALs).

In support of the three parts of the CC listed above, it is anticipated that other types of documents will be published, including technical rationale material and guidance documents.

3 KISEC in the Part 2 of ISO/IEC 15443

KISEC was introduced on the 10th Canadian IT Security Symposium 1998 and ICCC 2000, and will be the parts of ISO/IEC 15443 Part 2. Next is brief overview of KISEC, and the base of this research.

3.1 Aim

The purpose of KISEC is to provide a framework of security evaluation criteria and security evaluation methodology for firewall (FW) and intrusion detection system (IDS) in Korea.

3.2 Description

The evaluation criteria “Korea Information Security Evaluation Criteria (KISEC)” and the evaluation methodology “Korea Information Security Evaluation Methodology (KISEM)” were developed with three objectives in 1998:

- to provide a hierarchical rating scale for the evaluation of firewalls and intrusion detection systems;
- to provide a method for specifying trusted firewalls and intrusion detection systems in procurements;
- to cumulate 'know-how's related to IT security evaluation by operating its' own evaluation criteria and methodology.

The KISEC defines functional and assurance requirements for the each of seven evaluation levels (K1 ~ K7). Each level has a set of functional and assurance requirements to which evaluated firewalls and intrusion detection systems should conform. KISEC has some different functional requirements depending on product type such as firewalls and intrusion detection systems. However assurance requirements are commonly used for both firewalls and intrusion detection systems. The functional requirements are divided into some classes, such as Identification & Authentication, Integrity, Security Audit, Security Management, etc. Assurance requirements are divided into 6 classes, such as Development, Configuration Management, Testing, Operation Environment, Guidance Documents, and Vulnerability Analysis.

The specific level is determined according to the security functions implemented in the products and the confidence of assurance requirements. Depending on the security functional and assurance requirements, the evaluation level is divided into seven levels, from K1 to K7. K1 represents the lowest level and K7 the highest.

The followings are the characteristics of form K1 to K7 level:

Level K1 must satisfy the minimum level of security functions such as identification & authentication for system administrator and security management, etc. Also, there must be security target and functional specifications;
Level K2 must satisfy the requirements of the level K1 and be able to create and maintain audit records on security related activities. Also, architectural design document must be required. Vulnerability and misuse analysis of firewall or intrusion detection system must be carried out;
Level K3 must satisfy the requirements of the level K2 and be able to check whether there has been any modification to the stored data inside firewall or intrusion detection system and transmitted data. Also, detailed design and configuration management documents are required;
Level K4 must satisfy all the requirements of the level K3 and provide the identification & authentication function that protects firewall or intrusion detection system from replay attacks. Also, source code and/or hardware design documents are submitted;
Level K5 must satisfy all the requirements of the level K4 and provide mutual authentication function. Also, formal model of firewall or intrusion detection system security policy is required. Functional specifications, architectural design documents, and detailed design documents must be written in semi-formal;

Level K6 must satisfy the requirements of the level K5. At this level, consistency among detailed design documents, source code, and/or hardware design documents must be verified;
Level K7 must satisfy all the requirements of level K6. At this level, functional specifications and architectural design documents must be written in the formal so it is synchronized with formal model of system security policy.

The KISEM builds on the KISEC describing how firewalls and intrusion detection systems should be evaluated according to these criteria. The specific objective of the KISEM is to ensure that there exists a harmonized set of evaluation methods that complements the KISEC.

4 Relationship of Requirements for Configuration Management

In this paper, we will propose some results of our research for analyzing of relationship of configuration management requirements between KISEC and CC.

The reason of selection for the K4 level of KISEC is because the trial data of real evaluation for K4 in Korea are the most things. We analyze the meaning of phrases of KISEC and compare them with the description of assurance components and elements in the related assurance classes of CC. As a result of research, we are able to know similar and different things.

4.1 Configuration Management Requirements in KISEC

As mentioned earlier, KISEC consists of KISEC-Firewall and KISEC-IDS. Assurance requirements for configuration management of K4 level in KISEC-Firewall are described in K4.9 section (in KISEC-IDS, K4.11 section) as like;

228(English) The configuration management documentation, which includes the configuration list, the configuration identification method, and configuration management system must be provided for the evaluation of the configuration management.
229(English) The configuration list shall specify a unique version number that can identify firewall.
230(English) The configuration list should specify all components of firewall and unique identifiers for the following.

1. All components in firewall
2. All documentation produced in the development process
3. **Source code**
4. **Hardware drawings (if module is implemented in hardware)**

231(English) The configuration identification methods shall describe how the unique version number is given.

232(English) The configuration management system should describe how changes to the configurations are controlled.
233(English) The configuration management system shall ensure that the configuration management is correctly applied to every development process of firewall.
234(English) The configuration change control methods shall ensure that changes to the configuration items are only possible through authorized method.

4.2 Configuration Management Requirements in CC

Configuration management (CM) helps to ensure that the integrity of the TOE is preserved, by requiring discipline and control in the processes of refinement and modification of the TOE (Target of Evaluation) and other related information. CM prevents unauthorized modifications, additions, or deletions to the TOE, thus providing assurance that the TOE and documentation used for evaluation are the ones prepared for distribution.

CC contains ACM class for CM evaluation and ACM class contains next three families;

CM automation (ACM_AUT): Configuration management automation establishes the level of automation used to control the configuration items.
CM capabilities (ACM_CAP): Configuration management capabilities define the characteristics of the configuration management system.
CM scope (ACM_SCP): Configuration management scope indicates the TOE items that need to be controlled by the configuration management system.

Assurance requirements for configuration management in CC are described in ACM class and 3 families. To get the EAL3 (Evaluation Assurance Level 3) by evaluation, ACM_CAP.3 and ACM_SCP.1 components should be met (See Fig.1).

The component ACM_CAP.3 is consists of ten '.C' elements (evidence elements) described as like;

ACM_CAP.3.1C: The reference for the TOE shall be unique to each version of the TOE.
ACM_CAP.3.2C: The TOE shall be labeled with its reference.
ACM_CAP.3.3C: The CM documentation shall include a configuration list and a CM plan.
ACM_CAP.3.4C: The configuration list shall describe the configuration items that comprise the TOE.
ACM_CAP.3.5C: The CM documentation shall describe the method used to uniquely identify the configuration items.
ACM_CAP.3.6C: The CM system shall uniquely identify all configuration items.
ACM_CAP.3.7C: The CM plan shall describe how the CM system is used.

ACM_CAP.3.8C: The evidence shall demonstrate that the CM system is operating in accordance with the CM plan.
ACM_CAP.3.9C: The CM documentation shall provide evidence that all configuration items have been and are being effectively maintained under the CM system.
ACM_CAP.3.10C: The CM system shall provide measures such that only authorized changes are made to the configuration items.

Assurance Class	Assurance Family	Assurance Components by Evaluation Assurance Level						
		EAL1	EAL2	EAL3	EAL4	EAL5	EAL6	EAL7
Configuration management	ACM_AUT				**1**	1	**2**	2
	ACM_CAP	**1**	**2**	**3**	**4**	4	**5**	5
	ACM_SCP			**1**	**2**	**3**	3	3
Delivery and operation	ADO_DEL		**1**	1	**2**	2	2	**3**
	ADO_IGS	**1**	1	1	1	1	1	1
Development	ADV_FSP	**1**	1	1	**2**	**3**	3	**4**
	ADV_HLD		**1**	**2**	2	**3**	**4**	**5**
	ADV_IMP				**1**	**2**	**3**	3
	ADV_INT					**1**	**2**	**3**
	ADV_LLD				**1**	1	**2**	2
	ADV_RCR	**1**	1	1	1	**2**	2	**3**
	ADV_SPM				**1**	**3**	3	3
Guidance documents	AGD_ADM	**1**	1	1	1	1	1	1
	AGD_USR	**1**	1	1	1	1	1	1
Life cycle support	ALC_DVS			**1**	1	1	**2**	2
	ALC_FLR							
	ALC_LCD				**1**	**2**	2	**3**
	ALC_TAT				**1**	**2**	**3**	3
Tests	ATE_COV		**1**	**2**	2	2	**3**	3
	ATE_DPT			**1**	1	**2**	2	**3**
	ATE_FUN		**1**	1	1	1	**2**	2
	ATE_IND	**1**	**2**	2	2	2	2	**3**
Vulnerability assessment	AVA_CCA					**1**	**2**	2
	AVA_MSU			**1**	**2**	2	**3**	3
	AVA_SOF		**1**	1	1	1	1	1
	AVA_VLA		**1**	1	**2**	**3**	**4**	4

Fig. 1. Evaluation assurance level summary

And the component ACM_SCP.1 is consists of two '.C' elements described as like;

ACM_SCP.1.1C: The CM documentation shall show that the CM system, as a minimum, tracks the following: the TOE implementation representation, design documentation, test documentation, user documentation, administrator documentation, and CM documentation.
ACM_SCP.1.2C: The CM documentation shall describe how configuration items are tracked by the CM system.

But in this paper, only ACM_CAP.1 component is considered.

4.3 Comparison between KISEC and CC

In this paper, the research for compliance is done only about the ACM_CAP.3. ACM_CAP.3.1C and ACM_CAP.3.2C require that the TOE shall be labeled with its unique reference. These requirements may be met by 229th phrase of KISEC, and the result of compliance comparison is described "Compliance."

But ACM_CAP.3.3C requires that a configuration list and a CM plan shall be included in the CM documentation. But the requirements of KISEC do not mention about the CM plan. In the real case, CM plan may be used, but the requirements do not need that explicitly. 228th phrase of KISEC require configuration list and other things but not CM plan. Therefore, in this case, the result of compliance comparison is described "Partial."

The requirement of ACM_CAP.3.4C is met by 230th phrase, ACM_CAP.3.5C is met by 231st phrase, ACM_CAP.3.6C is met by 230th phrase of KISEC, and therefore, the results are "Compliance."

ACM_CAP.3.7C requires a thing under the assumption CM plan exists. But the KISEC does not need the CM plan, and therefore, in this case, the comparison is not proper and the result is "None."

The requirement of ACM_CAP.3.8C may be met by 233rd phrase, but the problem about the CM plan exists. So the comparison is not proper and the result may be "None."

ACM_CAP.3.9C and ACM_CAP.3.10C are met by 232nd, 233rd, 234th phrases of KISEC and the results are "Compliance."

The results are described in Fig.2 as below.

Requirements of K4		Requirements of ACM_CAP		
		Compliance	Deficiency	
			Partial	None
K4.11	228		ACM_CAP.3.3C	ACM_CAP.3.7C ACM_CAP.3.8C
	229	ACM_CAP.3.1C ACM_CAP.3.2C		
	230	ACM_CAP.3.4C ACM_CAP.3.6C	ACM_CAP.3.2C	
	231	ACM_CAP.3.5C		
	232	ACM_CAP.3.10C		
	233	ACM_CAP.3.9C		
	234	ACM_CAP.3.10C		

Fig. 2. Compliance of K4.9 with EAL 3

5 Conclusions and Future Work

The objective of this paper is to contribute research data to ISO/IEC 15443 Part 3. The material in Part 3 targets the IT security professional who must select assurance methods and approaches. This paper is a part of research for the relationship of assurance requirements between KISEC and CC. The contents of this paper may be included in ISO/IEC 15443 Part 3, and will help the development company of IT product and system to understand the evaluation criteria CC and prepare for an evaluation.

In this paper, only the requirements for configuration management are described for compliance between KISEC and CC. In facts, there were many evaluation criteria in the world, and they were merged into CC. Therefore, the evaluations for IT products and systems now have new ages, and for the future, the evaluations will be carried out by CC. But the evaluation results carried out in the past by the criteria each nation's such as ITSEC, TCSEC, KISEC, CTCPEC and others should be sustained via the research for the compliance between new criteria and old criteria. This paper may be the start and the part of that research. For the future, more detail researches are needed.

References

1. ISO. ISO/IEC 15408-1:1999 Information technology - Security techniques - Evaluation criteria for IT security - Part 1: Introduction and general model
2. ISO. ISO/IEC 15408-2:1999 Information technology - Security techniques - Evaluation criteria for IT security - Part 2: Security functional requirements
3. ISO. ISO/IEC 15408-3:1999 Information technology - Security techniques - Evaluation criteria for IT security - Part 3: Security assurance requirements
4. ISO. ISO/IEC 15443-1 Information technology - Security techniques - A framework for IT security assurance - Part 1: Overview and framework
5. ISO. ISO/IEC 15443-2 Information technology - Security techniques - A framework for IT security assurance - Part 2: Assurance methods
6. ISO. ISO/IEC 15443-3 Information technology - Security techniques - A framework for IT security assurance - Part 3: Analysis of assurance methods
7. KISA. Information Security Systems & Certification Guide, 2002
8. ISO. ISO/IEC WD 18045 Methodology for IT Security Evaluation

Term-Specific Language Modeling Approach to Text Categorization*

Seung-Shik Kang

School of Computer Science, Kookmin University & AITrc, Seoul 136-702, Korea
sskang@kookmin.ac.kr

Abstract. In the probabilistic model of text categorization, we assume that terms are characterized by their statistical distribution of *tf-idf* metrics. However, we feel that classical *tf-idf* metrics may not be the best method solution for information retrieval and text categorization. We explored a language modeling approach with term-specific weighting method to improve the performance of text categorization system. To make our method comparable to the previous approaches, we performed an experiment and compared it to basic models. Term-specific language modeling approach to text categorization problem significantly outperformed the baseline model on each point of the evaluation.

1 Introduction

Text categorization has become one of the important techniques for handling and organizing a large volume of text data. The goal of text categorization is to classify the documents into a fixed number of predefined categories. One of the most commonly investigated applications of text categorization is a topic spotting for news articles. The current researches on text categorization have been focused on the two major issues: the method of computation by classifier models, and the extraction of learning features [1,2,3]. For classifier models, there are statistical approaches, machine learning models, and information retrieval models. As for the representation method of categories and input documents, classifiers are the representatives of document vectors and the most significant feature of classifiers is the weighting value of terms in the category.

The first step to text categorization is an extraction of classifiers from the document sets, which typically are a sequence of words or terms [4,5]. And then, features of classifiers are learned by classifier model. In statistical approaches, term frequency and inverted document frequency are used to calculate the weight values of classifiers, and the features are automatically trained in machine learning models [6,7,8,9]. Feature learning system is based on the terms and term frequencies. Because of the com-

* This work was supported by the Korea Science and Engineering Foundation (KOSEF) through the Advanced Information Technology Research Center(AITrc).

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 735–742, 2004.
© Springer-Verlag Berlin Heidelberg 2004

mon terms of high frequency, both *tf* and *idf* metrics are combined as a term weighting scheme. Though frequency-based metrics are commonly used as term weighting scheme in information retrieval systems, it is not sufficient for the representation of the text. Also, in the text categorization, the efforts to improve the performance of the categorization algorithm have a limitation if we insist to use the frequency-based weighting scheme.

Text categorization have been extensively studied and most of them are based on the probabilistic model of using *tf-idf* weighting scheme. One of the important components of text categorization model is to identify some useful terms in the document and estimating the term probabilities on the document. However, the standard *tf-idf* estimation of term probabilities has a limitation and it will not lead to better performance of the text categorization system. In this paper, we applied a language modeling approach with a term-specific weighting method to the text categorization problem.

2 Related Work

A large number of classification models have been explored. They are nearest neighbour methods, Naïve Bayesian classifiers, decision trees, neural network models, and support vector machines. Naïve Bayes (NB) classifiers have exhibited good performance in recent studies and a good number of NB methods are published. The basic idea of NB starts from the theory of conditional probability that estimates the similarity of the given document to the predefined categories. P(t|c) is the maximum likelihood estimation of the probability of term *t* in category *c*.

$$P(c \mid d) = \frac{P(c)P(d \mid c)}{P(d)} \approx P(c) \times \prod_{t \in d} P(t \mid c)^{tf(t,d)}$$

$$P(t \mid c) = \frac{tf(t,c)}{\sum_{i} tf(t_i,c)}$$

There are several versions of NB classifiers. One of them is a multiple Bernoulli model that a document is treated as a binary vector of terms. Another NB model is a multinomial model. It treats the document as a sequence of independent terms, rather than a binary vector. SVM is claimed to get a high precision [10]. However, it is not clear which model is the best one because of the differences in test data. Yang(1999) notes that SVM, kNN and LLSF(linear least squares fit) outperformed neural net and NB for small training instances per category, but all the methods perform comparably when the training data are sufficiently common [1]. For the text categorization task, classifiers and features are determined for each category, assigning weight values on them. Basic discriminative features of the categorization are term frequency and inverted document frequency. This metric is applied as a key factor for the feature se-

lection methods of χ^2 statistics, mutual information, expected mutual information, and information gain. Relevance model in classical information retrieval follows the 'probability ranking principle' that estimates the probability of a document for the relevant set [11]. Robertson(1997) asserts that probability D for relevant set R is estimated by P(D|R) / P(D|N). If we take a word independence assumption, then the ranking of documents are calculated by the following equation.

$$\frac{P(D \mid R)}{P(D \mid N)} \approx \prod_{t \in D} \frac{P(t \mid R)}{P(t \mid N)}$$

Recent works in information retrieval has been shifted from the classical probabilistic model (relevance model) to the language model [12,13,14]. Ponte(1998) inferred a language model for each document and ranked it according to the estimation of producing the query the document model. The query generation probability $P(Q \mid M_d)$ is formalized either as a multiple Bernoulli model or a multinomial model.

$$P(Q \mid M_d) = \prod_{t \in Q} P(t \mid M_d) \prod_{t \notin Q} (1 - P(t \mid M_d))$$

Language model estimates a generation probability of term *t* by document model M_d as a function of the maximum likelihood estimator, the average probability of term *t* in the collection, and the risk function R(t, d). In the equation $P(t|M_d)$, $P_{ml}(t,d)$ is the maximum likelihood estimation of term *t* in document *d*. $P_{avg}(t)$ is an average of $P_{ml}(t,d_i)$ for all d_i in the collection and the risk function R(t,d) is a geometric distribution. P(t|G) is collection probability of *t*, where collection frequency of term *t* is divided by total number of tokens in the collection.

$$P(t \mid M_d) = \begin{cases} p_{ml}(t,d)^{1-R(t,d)} \times p_{avg}(t)^{R(t,d)} & \text{if } tf(t,d) > 0 \\ P(t \mid G) & \text{otherwise} \end{cases}$$

$$R_{t,d} = \frac{1.0}{1.0 + \bar{f}_t} \times \left(\frac{\bar{f}_t}{1.0 + \bar{f}_t} \right)^{tf_{t,d}}$$

The essential point of language model is the smoothing of probabilities [15]. Term probabilities are estimated by average probabilities in other documents and collection probabilities together with the standard *tf-idf* metric. The smoothing technique plays an important role to estimate the underlying probabilities of terms for query generation model. Other smoothing technique of linear interpolation works well for the relevance-based language model [14]. Lavrenko(2001) used a linear interpolation technique to smooth the maximum likelihood probability with collection probability. Smoothing parameter λ is decided by experiment as a simple constant.

$$P(t \mid M_c) = \lambda\, p_{ml}(t,c) + (1 - \lambda) P(t \mid G)$$

3 Term-Specific Language Model for Text Categorization

Standard probabilistic model for text categorization assigns a category for a given document by the similarity measure between input document and features of categories. Instead, language modeling approach to text categorization estimates the document generation probabilities. In the text categorization model, documents are treated as a sequence of independent terms by 'term independence assumption'.

3.1 Language Model for Text Categorization

For automatic text categorization, features are extracted from the training corpus and parameters are estimated by learning process. In the previous approaches, parameters are estimated by *tf-idf* statistics. Instead, language modeling approach combines three metrics of average probability, collection probability, and maximum likelihood probability.

As a parameter learning method in language modeling approach to text categorization, term probabilities are estimated mainly from the maximum likelihood estimator, but average probability is combined to get a more accurate estimation of parameters. For term t in the training corpus of category c, $P(t|M_c)$ is calculated by $p_{ml}(t,c)$ and $p_{avg}(t)$ with a risk factor $R(t,c)$. As a smoothing technique for terms that are non-existent in the training corpus of category c, the collection probability $P(t|G)$ is assigned. $P(t|G)$ is a collection frequency over the size of the collection.

$$P(t \mid M_c) = \begin{cases} p_{ml}(t,c)^{1-R(t,c)} \times p_{avg}(t)^{R(t,c)} & if\ tf(t,c) > 0 \\ P(t \mid G) & otherwise \end{cases}$$

Language model in information retrieval attempts to model the query generation process. In the text categorization problem, category assignment probability is estimated as a document generation process by predefined categories. Document generation probability for input document D by category c is estimated as a product of term probabilities according to multinomial view of the model and term independence assumption. Automatic text categorization is to assign a category with the highest probability and the category of the maximum probability is assigned to document D.

$$x = \arg\max_{c} P(D \mid M_c)$$

$$P(D \mid M_c) = \prod_{t} P(t \mid M_c)^{tf(t)}$$

3.2 Term-Specific Text Categorization

In standard probabilistic models, we assume that the probabilities of terms are determined only by their frequency statistics in the document. This assumption may not be the best solution of document representations that we are apt to fall into an error that all the nouns in a document have the same roles in the document. For the weighting scheme of terms, there are two points of views for document representation: (1) a discriminative value that distinguishes or characterizes the document from others; (2) an importance measure as a keyword or a stopword [16].

Another weighting scheme is a content-based term weighting that is based on the term importance factors in a document. It is an analytic approach that analyses the contents of a document to evaluate terms in the document. The weight value of a term is calculated by importance factors as a keyword in a document. Term weighting depends on several factors such as the type of a document, the relative location in the document, and the role of terms in a sentence or a paragraph [17].

Thematic words of a document are representative terms for the document. Thematic words are extracted from a text by analysing the contents of the text. Most of keywords are found in the title or an abstract in a research paper that consists of a title, abstract, body, experiment, and conclusion. Also, newspaper article contains a keyword in the title or the first part of text. There are some clues of determining a keyword and we may classify the types as word level, sentence level, paragraph level, and text level features. Word-level features are part-of-speech and case-role information.

Syntactic or sentence-level features are the type of a phrase or a clause, sentence location, and sentence type. From the rhetoric word in a sentence, the importance of the sentence is computed and the terms in a sentence are affected by the sentence type. Also, the weight of terms in subjective clause is not equal to the same term that occurs in auxiliary clause or modifying clause. Basic term weight is assigned by the type of a term and it is recomputed by the features that it accompanies in the text. That is, the weight value of a term is also determined by the characteristics of word, sentence, phrase, and clause where the term is extracted.

Term weighting scheme is applied to the document generation process in language modeling approach. On the assumption that the importance factor of terms in a document is distinguishable, we adjusted term probabilities that are estimated by language model. For each term probabilities, we multiplied the term weight $w_D(t)$ for input document D. As a result, document generation probability is estimated by the following equation.

$$P(D \mid M_c) = \prod_t (w_D(t) \times P(t \mid M_c))^{tf(t)}$$

4 Experimental Results

We performed experiments on language modeling approach and term weighing methods for text categorization. In addition, we tried to find out the performance determining factors to get a better solution of the text categorization problem. We evaluated our method for text categorization with feature selection by χ^2 statistics. Base-line model is a Naïve Bayesian classifier and the test data in our experiment are news group data collection. This data collection consists of about 10,331 documents for 15 categories [18]. We used 7,224 articles for training data and 3,107 articles for test data.

Table 1. Performance evaluation in F-measure

Model # terms	NB	LIS	LM	T-LIS	T-LM
10000	0.714	0.788	0.818	0.801	0.822
20000	0.733	0.804	0.838	0.815	0.842
30000	0.746	0.819	0.844	0.829	0.855
40000	0.755	0.828	0.853	0.837	0.862
50000	0.767	0.835	0.861	0.845	0.868
60000	0.777	0.841	0.866	0.853	0.871
70000	0.791	0.844	0.863	0.857	0.871
80000	0.804	0.857	0.870	0.870	0.880

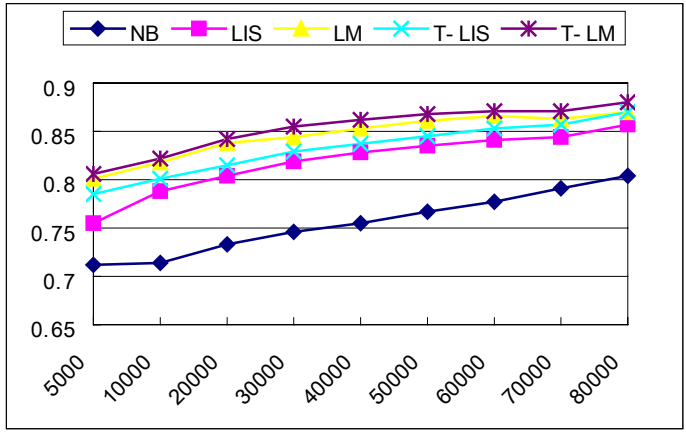


Fig. 1. Performance evaluation for text categorization

In the experiments, about 90,000 features have been extracted from the training data and we pruned the terms with low probabilities by χ^2 statistics. Table 1 shows

the experimental results. Five different methods are shown in this table: (1) Naïve Bayes' model(NB), (2) linear interpolation smoothing(LIS), (3) language modeling approach(LM), (4) linear interpolation smoothing with term weighting method(T-LIS), and (5) language modeling approach with term weighting method(T-LM).

The result shows that language modeling approach achieved about 7%(LM) or 8%(T-LM) improvements when compared to the baseline system of Naïve Bayes' model. Term weighting methods of T-LIS and T-LM shows about 1% improvements when compared to LIS and LM. Figure 1 shows the performance evaluation as a graph and we see that the performance is monotonically increasing according to the number of features. The contribution of content-based term weighting for input document has been evaluated to be about 1% improvement of the overall performance. It depends on the accuracy of document analysis technique and we expect that the performance will be better by deep analysis of the document.

5 Conclusion

It is common that information models are based on the probability estimation frequency-based metrics that is one of the important factors for term weighting. In this paper, we proposed a new method of applying language modelling approach to text categorization problem that considers average term probability and collection probability as a smoothing technique. Also, we used a term weighting method that is performed through the content analysis of document. The weighting scheme of terms is based on the term importance measures in a sentence or a paragraph. Experimental results show that our method of applying language model and term weighting scheme significantly improved the overall performance when compared to the baseline method. Language modeling approach achieved about 7% improvements and term weighting method achieved additional 1% improvements.

References

1. Yang, Y., Liu, X.: A Re-examination of Text Categorization Methods. Proceedings of Int. Conference on Research and Development in Information Retrieval (1999), 42-49
2. Yang, Y., Pedersen, J. P.: A Comparative Study on Feature Selection in Text Categorization. Jr. D H. Fisher(eds.), Proceedings of the 14th Int. Conference on Machine Learning (1997) 412-420
3. Cohen, W., Singer, Y.: Context-Sensitive Learning Methods for Text Categorization. ACM Transactions on Information Systems, Vol.17, no.2 (1999) 141-173
4. Rijsbergen, C., Harper, D., Porter, M.: The Selection of Good Search Terms. Information Processing and Management, Vol. 17 (1981) 77-91
5. Mladenic, D., Grobelnik, M.: Feature Selection for Unbalanced Class Distribution and Naive Bayes. Proceedings of the 16th International Conference on Machine Learning, Morgan Kaufmann (1999) 258-267

6. Sebastiani, F.: Machine Learning in Automated Text Categorization. ACM Computing Surveys, Vol.34, no.1 (2002) 1-47
7. Dumais, S., Platt, J., Heckerman, D., Sahami, M.: Inductive Learning Algorithms and Representations for Text Categorization. Proceedings of the 7th International Conference on Information and Knowledge Management (1998) 148-155
8. Lewis, D., Ringuette, M.: A Comparison of Two Learning Algorithms for Text Categorization. Proceedings of SDAIR-94: 3rd Annual Symposium on Document Analysis and Information Retrieval (1994) 81-93
9. Cohen, W.: Learning to Classify English Text with ILP Methods. Proceedings of the 5th International Workshop on Inductive Logic Programming (1995) 3-24
10. Joachims, T.: Text Categorization with Support Vector Machines: Learning with Many Relevant Features. European Conference on Machine Learning (1998) 137-142
11. Robertson, S.: The Probability Ranking Principle in IR, Morgan Kaufmann Publishers (1997) 281-286
12. Ponte, J., Croft, W.B.: A Language Modeling Approach to Information Retrieval. Proceedings of the 21st ACM SIGIR'98 (1998) 275-281
13. Song, F., Croft, W.B.: A General Language Model for Information Retrieval. Proceedings of the 22nd ACM SIGIR'99 (1999) 279-280
14. Lavrenko, V., Croft, W.B.: Relevance-Based Language Modes. Proceedings of ACM SIGIR'01 (2001) 120-127
15. Yang, Y.: A Study on Thresholding Strategies for Text Categorization. Proceedings of SIGIR'01 (2001) 137-145.
16. Lai, Y., Wu, C.: Meaningful Term Extraction and Discriminative Term Selection in Text Categorization via Unknown-Word Methodology. ACM Transactions on Asian Languages Information Processing, Vol. 1, no.1 (2002) 34-64
17. Kang, S., Lee, H., Son, S., Hong, G., Moon, B.: Term Weighting Method by Postposition and Compound Noun Recognition. Proceedings of the 13th Conference on Korean Language Computing (2001) 196-198
18. Ko, Y., Park, J., Seo, J.: Automatic Text Categorization using the Importance of Sentences. Journal of Korean Information Science Society: Software and Application (2001) 417-423

Context-Based Proofreading of Structured Documents*

Won-Sung Sohn, Teuk-Seob Song, Jae-Kyung Kim, Yoon-Chul Choy,
Kyong-Ho Lee, Sung-Bong Yang, and Francis Neelamkavil

Department of Computer Science, Yonsei University,
Shinchon-dong, Seodaemun-ku, 120-749, Seoul, Korea
{sohnws, teukseob, ki187cm, ycchoy}@rainbow.yonsei.ac.kr
{khlee, yang, francis}@cs.yonsei.ac.kr

Abstract. To produce accurate editing results, the ambiguity of editing scopes related to marked correction signs should be solved. Proofreading the web document modifies the document structures, and the modified structures should be robustly valid for the defined DTD. This paper presents a pen-based proofreading interface in the XML document. In the proposed interface, correction signs are free-drawn, and the editing scopes are recognized and revised based on the contexts of the document to minimize the ambiguity of the editing scopes. The proposed interface provides both implicit and explicit modification methods for document structures. As a result, the editing scopes processed in the proposed interface are more accurate, and the document structures are maintained valid for DTD after the editing.

1 Introduction

Due to the development of the web technologies, now we have proofreading systems that allow interaction between the proofreader and the author in the on-line environment[1],[2],[3],[4],[5]. Compared to the proofreading of the traditional paper document, online proofreading provides more advantages such as simple editing processes, free editing and drawing of the correction marks, collaboration by multiple users, and reusing of the correction results,[1],[2],[3],[4],[5],[6],[7],[8],[9]. In the current proofreading interface, the standard correction marks used in the traditional paper document are drawn with a pen device,[2],[7],[9],[10].

There are PenEdit[6] and MATE[10] as the systems that support the pen-based proofreading model. In MATE, gestures are drawn with a stylus and properly translated for editing through a special recognizer. PenEdit allows pen-based marking of editing signs on the screen with a gesture recognition method. To provide a proofreading system that uses digital inking method in the electronic document, technical problems should be solved first. In the pen-based interface, there exist an ambiguity problem with regard to identifying free-drawings by the user and the related editing scopes. Because of the ambiguity, sometimes the editing results are different with the

* This work was supported by the IITA International Joint Research Project of 2003.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 743–753, 2004.
© Springer-Verlag Berlin Heidelberg 2004

editor's intentions, as shown in Figure 1. However, many of the current pen-based marking systems do not yet analyze nor resolve the ambiguity problem[4],[7],[9],[10],[11].

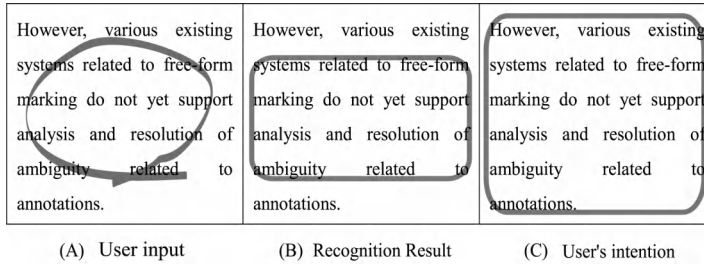


Fig. 1. Example of the editing scope ambiguity.

In Aamaya+PEN[7], the proofreading target is the HTML document unlike in other proofreading interfaces described above. Aamaya+PEN provides pen-based proofreading of the web document with an HTML editor called Amaya[12]. However, the ambiguity problem also exists in this system. The system uses only simple methods of inserting and deleting the <P> tag at modifications of document structures in editing the HTML document. Now the XML document, which has explicit document structures, should be included in the editing target[13].

In this work, we designed and implemented a proofreading interface based on the XML document. In the proposed interface, correction marks are free-drawn, and context-based scope recognition and revision methods are provided to minimize the ambiguity of the editing scopes related to the marked correction signs. The interface provides both implicit and explicit modification methods for document structures. The structures are automatically maintained by the implicit modification method at modifications of contents and structures by the marked correction signs. The structures are maintained by the user interaction by the explicit modification method. As a result, the proposed proofreading interface results in more accurate editing scopes than other systems and maintains document structures valid for DTD of the original document at their modifications by the marked correction signs.

2 Overview of the Proposed Interface

In this section, we review the overall processing of the proposed proofreading interface. The proposed interface provides two kinds of the user interfaces for marking the correction signs, as shown in Figure 2. They are a free-form drawing method and a dragging method. The first method focuses on easiness, whereas the second one emphasizes preciseness. We valued both easiness and preciseness in the proofreading system on the following grounds. First, the drawing interface that employs a pen or a mouse rather than menus as its device for inserting correction marks provides the same environment as editing the paper document. Second, in addition to the easiness,

for preventing the mistakes frequent in the pen-based interface, correct editing scopes should be determined by the user.

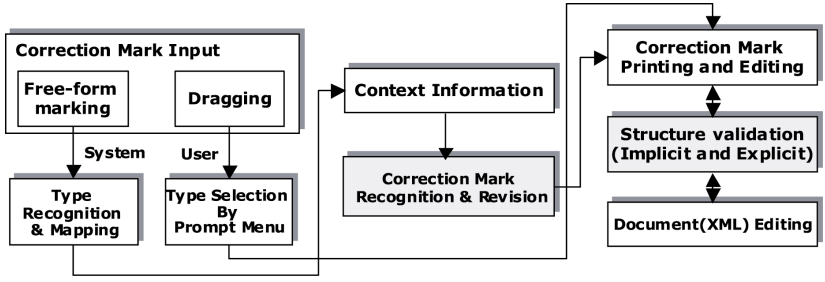


Fig. 2. Proposed pen-based proofreading interface.

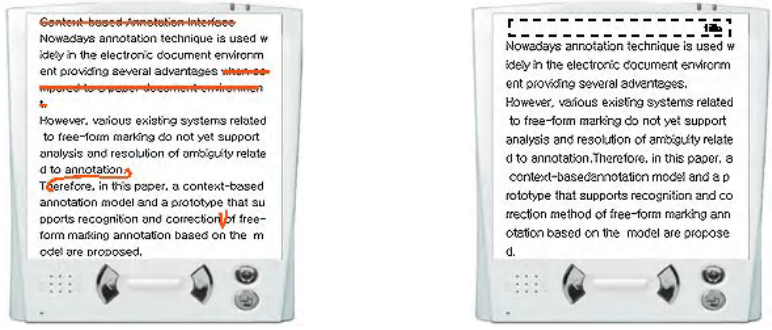
This paper defines 12 types of correction marks, as shown in Table 1. The standard correction marks called Chicago Style Proofreader's Marks[14] were simplified. The types of the correction marks created by the free-form drawing method are identified by the feature-based gesture recognition method[15].

Table 1. The types of the correction signs used in this paper.

Correction	Notation	Correction	Notation
Delete		CloseUp	
Insert		MoveLeft	
Transpose		MoveRight	
SplitPara		CreateLink	
JoinPara		DeleteLink	
Replace		Note	

The final scopes of the marked signs are displayed on a screen after they are automatically revised by the context-based recognition and revision method, which solves the ambiguity problem. Then the displayed correction marks can be directly used for editing or remain displayed for selective or lump-sum corrections of the document at the user's execution command by a menu. Lastly, the structures of the document are maintained valid for DTD by the implicit and explicit structure modification methods at modifications of the contents and structures of the XML document due to marked correction signs. The document structures are automatically maintained by the implicit method, whereas they are maintained by the user interaction in the explicit method. The dragging method will not be discussed any more because the recognition and scope determination proposed in this paper are not performed in the method.

Figure 3 shows actual marking of correction signs and correction results. Figure 3(A) shows 4 markings of correction signs, and Figure 3(B) is the results of applying each of the correction commands.



(A) (B)

Fig. 3. Markings(A) and editing results(B) in the proposed system.

As the first delete mark includes the structure that should definitely occur in DTD, the related scope is separately displayed even after deletion was executed. The scopes of other correction marks are revised by the recognition module, and corrections are executed as shown in Figure 3(B). The detailed description of the proposed method is as follows.

3 Editing Scope Revision and Structure Modification

This section describes context-based scope recognition and revision methods which minimize the ambiguity that occurs when the system determines the relations between the on-line correction marks and target texts. In addition, this paper proposes a method that maintains DTD validation of document structures even after their modifications in the editing processes.

3.1 Context-Based Recognition and Revision Methods for Editing Scopes

The proposed method for recognizing and revising the editing scopes is processed by rule-based models composed of conditional clauses. In this paper, a set of 120 rules was established for recognition of 12 types of correction marks and editing scopes. The relations between the correction marks and text scopes are presented as contextual sectors as shown in Figure 4 for recognition and revision of the editing scopes. Each of the proposed sectors is composed of contextual units of letters, words, sentences, and paragraphs, and each unit is divided by a box. The internal scope of the box is divided by vertical and horizontal lines, or top and bottom lines on the units of the sectors. The proposed recognition and revision methods include different units of

the sectors depending on the types of marked correction signs. For instance, the scope for a 'delete' mark is recognized by letter, word, and sentence units, whereas the scope for a 'joinPara' mark is recognized by a paragraph unit. This section will mostly use the 'delete' marking as examples to explain the proposed method.

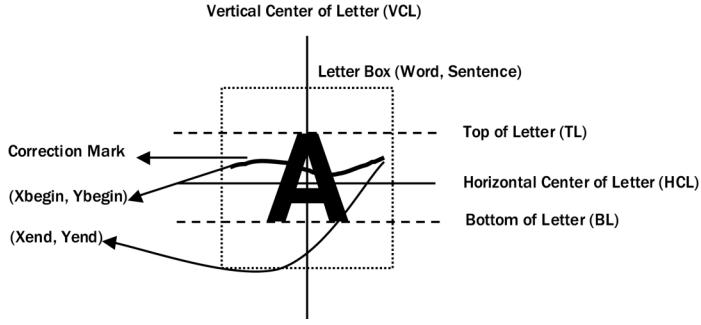


Fig. 4. Contextual sectors for the analysis of the editing scopes.

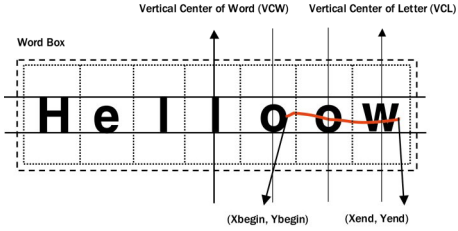


Fig. 5. Scope analysis on the letter level.

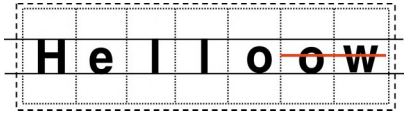


Fig. 6. Display of results of the scope revision in Figure 5.

Rule (23) ;
IF : (1) A correction mark type is *delete*.
(2) The length of the recognized X_{begin} and X_{end} is above the threshold of word area.
THEN : (1) The editing scope is a set of word boxes.
(2) Follow the next rule.

Revision of the Editing Scopes on the Letter Level. In Figure 5, the 'delete' mark is contacted to the 6th 'w' and the 7th 'o' starting from the left side of the letter boxes, but its contact to the 5th 'o' is not clear. In this case, inclusion of the letter in the ambiguous position as the editing target depends on whether X_{begin} and X_{end} is located on the left or the right of VCL of the letter box. In the example of Figure 5, as to the 5th 'o', X_{begin} of the correction mark is located on the right of VCL, therefore the letter is included in the editing scope by the proposed rule, and then the next letter box is analyzed. X_{end} is located on the right of VCL of the last letter box, therefore the last 'w' is included in the editing target. By this way, the editing scope in Figure 5 is revised to include the 6th 'o' and the 7th 'w', which is illustrated in Figure 6. The length of the scope of the letter boxes between X_{begin} and X_{end} in Figure 6 includes a small portion of the word box composed of the letter boxes. Because Figure 6 does not satisfy Rule(23) or Equation(1), the editing command is executed on the letter level which is

a contextual editing unit included in the scope of a word. If the example in Figure 6 satisfies the threshold of word area such as Rule(23) or Equation(1), the command should be executed not on the letter level but on the word level, and the editing scope should be revised accordingly.

$$ThWA = \frac{\sum Total\ Lengthof\ Xmin And\ Xmax of\ The Correction Mark}{\sum Total\ Lengthof\ Word\ Box\ Area} \quad (1)$$

Revision of the Editing Scopes on the Word Level. Figure 7 shows an example of a 'delete' command on the word level, where more than one words are the editing targets. The proposed method analyzes relations between the correction marks and word boxes to extract editing scopes of the marks. In Figure 7, the scope between X_{begin} and X_{end} of the 'delete' mark does not pass a sentence box, whereas it includes more than one word boxes. Therefore, the proposed method considers that its editing unit is on the word level and looks for the first and the last of the word boxes. As for the first word box in Figure 7, X_{begin} is located on the left of VCW, therefore the word box is included in the editing target by the proposed rule. In the last word box, X_{end} is located on the left of VCW, therefore the word box located on the left of the concerned word box is included in the editing scope. By this way, the editing scope is revised to include the word boxes from 'this' and 'to'.

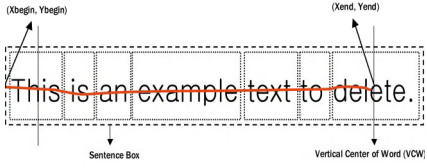


Fig. 7. Analysis of the editing scope of 'delete' command on the word level.

Fig. 8. Display of results of the scope revision in Figure 7.

In the example of Figure 7, it can be assumed intuitively that the editing scope includes the whole sentence scope rather than some words. Therefore, the editing scope extracted by the proposed rule, which is from 'this' and 'to', should be extended to the scope between 'this' and 'delete'. The proposed method revises the editing scope from the word level to the sentence level as shown in Figure 8 if the proposed threshold of sentence area is satisfied.

Revision of the Editing Scopes on the Paragraph Level. The proposed method applies different recognition rules according to the properties of the types of the correction marks. Figure 9 is an example of editing on the paragraph level, where a marked correction sign is a 'joinPara' that commands to join Paragraph (a) and (b). The start point of the mark is located at the end of the last sentence in Paragraph (a), and the end point is located at the start of the second sentence in Paragraph (b). In this case, due to the ambiguity of the marking, it is unclear before which sentence in Paragraph (b) Paragraph (a) should be joined. Generally, when paragraphs are joined, adjacent previous and latter paragraphs are joined together. It is rare that one para-

graph is inserted in the middle of another paragraph by a 'joinPara' mark. Therefore, in the example of Figure 9, we have to assume that the user marked the correction sign with the intention to join Paragraph (a) and (b). With regard to correction marks such as 'joinPara', this paper applies the editing rule for commands on the paragraph level to understand the user's intentions. In the example of Figure 9, the command to join Paragraph (a) and (b) is executed by Rule(62), and its editing scope is revised as shown in Figure 10.

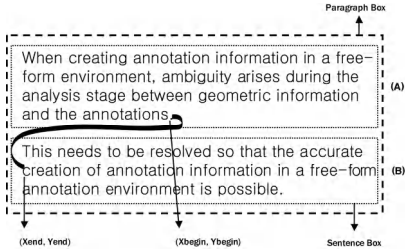


Fig. 9. Scope analysis on the paragraph level.

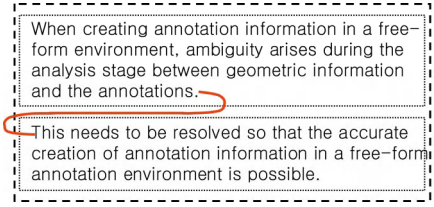


Fig. 10. Display of results of the scope revision in Figure 9.

Rule (62) ;

IF :	(1)	A correction mark type is 'joinPara'.
	(2)	$(X_{\text{begin}}, Y_{\text{begin}})$ and $(X_{\text{end}}, Y_{\text{end}})$ are located in different paragraph boxes.
	(3)	The paragraph boxes where $(X_{\text{begin}}, Y_{\text{begin}})$ and $(X_{\text{end}}, Y_{\text{end}})$ are located are adjacent from each other.
THEN :	(1)	The command is executed on the unit of paragraph boxes.
	(2)	Join paragraphs by extracting effective paragraph boxes.
	(3)	Follow the next rule.

3.2 Structure Modification Rule on Document Editing

Hypertext documents such as XML include structures unlike general text documents. When the document is edited, document structures modified due to execution of 'insert', 'delete', and other commands should remain valid for DTD of the document. This paper proposes a method that maintains DTD validation of document structures even after their modifications in the editing processes. For that purpose, implicit and explicit structure modification methods are proposed.

Implicit Structure Modification Method. In the proposed rule model on the editing of structure information, all the possible cases of modified structure information are extracted based on the recognized relations between the marked correction signs and structure information. Then the structure that is valid for the defined DTD is selected and automatically processed if structures can be properly modified by a system.

Figure 11(A) shows marking of a 'delete' command, and (B) shows an example of implicit modifications of document structures due to application of the command. If document structures are defined as 'title', 'author', and 'affiliate', the 'title' and the 'author' should definitely occur, whereas the 'affiliate' does not need to occur. There-

fore, in (1) and (2) of Figure 11(A), texts are deleted according to the occurrence indicator, but structures are maintained as shown in Figure 11(B). In Figure 11(A), (3), deleting nodes by the 'delete' command is not invalid for DTD, therefore the whole contents are deleted as shown in Figure 11(B).

Explicit Structure Modification Method. There could be many forms of structure information modified due to execution of an editing command, depending on the properties of the correction mark and positions of structure information. In that case, although the system can make determinations on its own and display them, the user might not be satisfied with the results. Therefore, this paper proposes an explicit method that the user himself can make choices by checking possible results on the screen.

Figure 12 shows how an 'insert' command is executed. Suppose that a document includes DTD structures as <!ELEMENT author (name, post?, position?. tel?, fax?, email?, homepage?, artwork?)>, the system cannot select one sub-structure among eight included in the 'author', the target structure for insertion. In that case, the proposed method displays all the possible sub-structures within the concerned scope in the list box form as shown in Figure 12(B). Then the user makes his own choice.

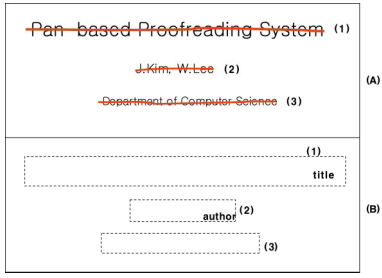


Fig. 11. Example of implicit structure modification.

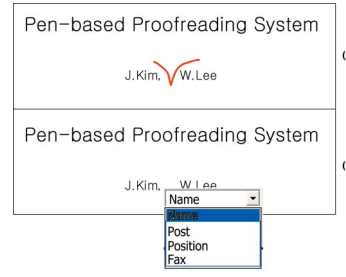


Fig. 12. Example of explicit structure modification.

The rule model for the explicit modification of structures is described in Rule(114).

Rule (114) ;			
IF :	(1)	A correction mark type is 'insert'.	
	(2)	The scope recognition task was performed.	
	(3)	There is structure information within the target scope.	
	(4)	There are more than one sub-element within the concerned element.	
THEN	(1)	The sub-elements in the concerned element are displayed in a list box.	
:	(2)	Insert the selected element and text.	
	(3)	Follow the next rule.	

4 Empirical Evaluation

In order to measure the accuracy of the recognized scopes on the correction signs marked by the user, we implemented editing prototypes where the proposed method and the method employed in Amaya+PEN[7] were applied and conducted ratings by the user. First, users were asked to draw free-form marks for 10 deletions, 10 replacements, and 10 insertions. And then they were shown the extracted scopes and asked to fill out question sheets that use scales from 1 (the lowest accuracy) to 10 (the highest accuracy) to compare the differences between the extracted scopes and the users' intended scopes. A group of 20 subjects consisted of 14 male and 6 female graduate students participated in the experiment.

An ANOVA with repeated measures was used to analyze the results. Figure 13 shows the users' subjective accuracy ratings on the marked correction scopes.

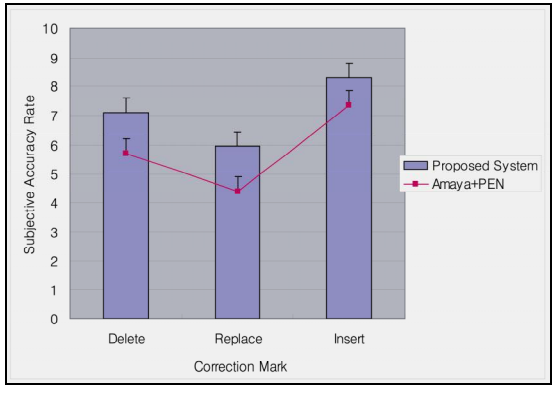


Fig. 13. Subjective accuracy ratings on the editing scopes.

As shown in Figure 13, the users rated that the results of the editing scopes in the proposed system were more accurate than those in the other system. Significant effects were found between the accuracy ratings for the proposed method and the other method ($F(1,114) = 36$, $P < 0.05$). Most of the users rated that the editing scopes determined by the proposed system were more accurate than those in the other system. In particular, higher ratings were given for the proposed system with regard to *delete* and *replace* commands. This is because those commands are mostly processed in more than one paragraph. It means that the proposed method is more effectively applied to commands of correction marks executed in multiple paragraphs.

5 Conclusions and Future Works

This paper proposed and implemented the interface that allows pen-based precise marking of correction signs in the XML-based electronic document. The interface

maintains document structures valid for DTD when the structures are modified due to editing.

The interface resolves the ambiguity of the editing scopes related to the marked correction signs by analyzing the properties of the pen-based correction marks and the contexts of the document.

The proposed interface provides both implicit and explicit structure modification methods. Document structures are automatically maintained by the implicit method, and the structures are maintained by the user interaction in the explicit method.

As a result, the editing scopes resulted by the proposed interface are more accurate than those of other systems, and document structures are maintained valid for DTD of the original document after the editing.

We plan to conduct more studies on collaboration works based on multiple users for document editing and on the management of document versions through modification detection of documents.

References

1. Blondin, S. & Buckingham, S. (1997). The future of online editing. International Broadcasting Convention (Conf. Publ. 447). IEE International, 38-45.
2. Farkas, D.K. & Poltrock, S.E. (1995). Online editing, mark-up models, and the workplace lives of editors and writers. IEEE Transactions on Professional Communication. 38 (2), 110–117.
3. Mori, D. & Bunke, H. (1997). Automatic Interpretation and Execution of Manual Corrections on Text Documents. HANDBOOK OF CHARACTER RECOGNITION AND DOCUMENT IMAGE ANALYSIS edited by Bunke & Wang. World Science Publishing Company.
4. Richy, H. & Lorette, G. (1999). On-line correction of Web pages. Document Analysis and Recognition. ICDAR '99. Proceedings of the Fifth International Conference. 581-584.
5. Zimmerman, D. & Yohon, T. (2002). An assessment of using online editing of students' assignments in an advanced technical writing class. Professional Communication Conference, IPCC 2002. IEEE International, 259-270.
6. Alexander, G. A. (1996). Applied Technology: A Closer Look at PenEdit. Seybold Report on Desktop Publishing. 8 (3), http://www.seyboldreports.com /SRDP/0dp8/D0803000.HTM.
7. ANDRE, J. & HELENE R. (1999). Paper-less Editing and Proofreading of Electronic Documents. Proc. of EuroTex'99.
8. Bunke, H., Gonin, R. & Mori, D. (1997). A tool for versatile and user-friendly document correction. Proceedings of Document Analysis and Recognition. 433-438.
9. Goldberg, D. & Goodisman, A. (1991). Stylus user interfaces for manipulating text. Proceedings of the fourth annual ACM symposium on User interface software and technology. South Carolina, ACM Press, NY, 127-135.
10. Hardock, G, Kurtenbach, G. & W. Buxton (1993). A Marking based Interface for Collaborative Writing. Proceedings of the sixth annual ACM symposium on User interface software and technology. Atlanta, ACM Press, NY, 259-266.

11. Gross, M. D. & Ellen Y. D. (1996). Ambiguous Intentions: A Paper-like Interface for Creative Design. Proceedings of the 9th annual ACM symposium on User interface software and technology. Seattle, Washington, 183-192.
12. Amaya (1997). http://www.w3.org/Amaya.
13. Sohn, W. S., et al., (2002). Standardization of eBook documents in the Korean Industry. Computer Standards & Interfaces. 24(1), 45-60.
14. University of Chicago Press. (1993). The Chicago Manual of Style. 14th edition. Chicago: University of Chicago Press.
15. Rubine, D. (1991). Specifying Gestures by Example. Computer Graphics. 25(4), 329-337.

Implementation of New CTI Service Platform Using Voice XML

Jeong-Hoon Shin[1], Kwang-Seok Hong[1], and Sung-Kyun Eom[2]

[1] School of Information and Communication Engineering, Sungkyunkwan University and SITRC, Suwon, Kyungki-do, 440-746 Korea
only4you@chol.com, kshong@skku.ac.kr
http://only4you.mchol.com, http://hci.skku.ac.kr
[2] DACOM R&D Center, Daejeon, Korea
skoem@dacom.net

Abstract. In this paper, we describe implementation of new CTI (Computer Telephony Integration) service platform with ASP (Application Service Provider) concepts and its applications. We developed new middleware and media processing tool, speech enable functions including signaling. And we also developed Voice XML interpreter. Our system emerged from traditional standard IVR platform. But, the concepts of the platform are quite different. Using this new CTI service platform, we divided service creation part from legacy telephony interface part. This approach gives us some advantages. First of all, web users can edit their service scenario. And, web user can handle service scenario more efficiently and flexibly. Finally, we proved the stability of application and its platform using results of service on commercial business using new platform. And we analyze strong point and weak point of new CTI service platform.

1 Introduction

CTI and Voice portal service fields are very important part of value-added service market in wired and wireless telecommunication [5, 6]. CTI market grows more 10 times than last 10 years. Fig 1 shows recent trends of CTI technology, in this figure, we can see integration of Internet technology.

Each carrier developed various service categories of intelligent service products. Services include collect-to-call, called-party billing, web-to-phone, multiparty call, third party billing service and so on [3].

We have planned to develop new nationwide representative-number ARS hosting service (CTI platform) based on the SS7 signaling [4]. Because, most of small and medium-sized enterprises do not have their own ARS system or call center, and also, big enterprise provides services using R2 signaling type and static call flow type [4].

In past and recent years, many solution vendors and carriers provided IVR (Interactive Voice Response) board service using fixed scenario. And the customers have to comply with the fixed scenarios [5, 6]. It was very inconvenient for customers to

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 754–762, 2004.
© Springer-Verlag Berlin Heidelberg 2004

modify and maintain their own service scenarios. But, the needs of changing their own scenarios grow more and more. Customers want to modify their own ARS service flows and audio files of announcement at right time. But, the architecture of traditional CTI service platform can't meet customer's needs.

Fig. 1. Recent trends of CTI technology

For these reasons, we proposed new service platform of CTI service, based on Voice XML standard [1]. In this new service platform, we separated application scenario part and media processing part. We developed telephony server, which includes only call flow control part and media processing control part. Application part is responsible for VXML script and CGI (Common Gateway Interface) [1, 2, and 3]. Therefore, suggested architecture of CTI platform can provides proper modification of menu trees and rapid modification of scenarios by customers. Services related in Caller ID were also enhanced, because signaling module convert R2 MFC to SS7 method [4].

Our paper is organized as follows. In section 2 we present the architecture of traditional CTI service platform. In section 3 we introduce new architecture of suggested CTI service platform. In section 4 we describe the specification of ARS hosting service. Using suggested architecture of CTI service platform, we implemented commercial ARS hosting service of nationwide. This section includes call flows of implemented ARS hosting service and implemented specification of VXML interpreter. In section 5 we analyze proposed new CTI service platform. And conclusions are given in section 6.

2 Architecture of Traditional CTI Service Platform

Fig.2 shows the architecture of traditional CTI service platform. In this architecture, CTI service platform includes service scenario handling parts. For this reason, customers can't modify their own service scenarios and solution vendors can't comply with various needs of CTI service markets.

Customers want to change their own service scenarios, ARS prompt, and menu trees with ease. These requirements made changes of architecture of traditional CTI service platform.

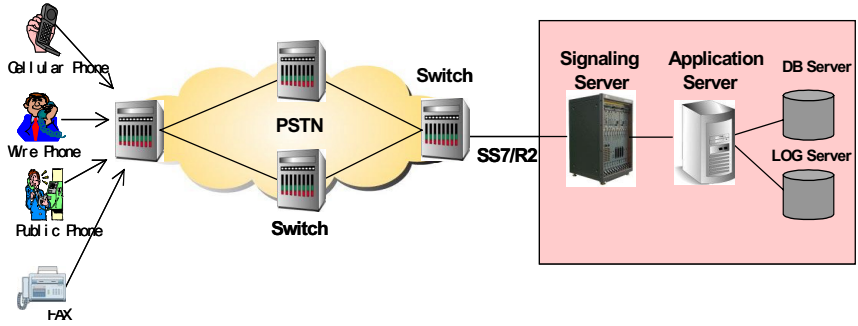


Fig. 2. Architecture of traditional CTI service platform

3 New Architecture of Suggested CTI Service Platform

Fig 3 shows new architecture of suggested CTI service platform. Figure 3 can be separated in two distinctive part, access server part and service scenario part.

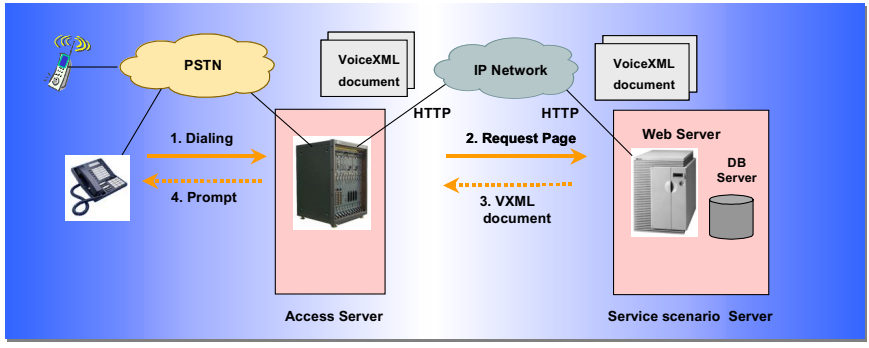


Fig. 3. Architecture of suggested CTI service platform

Access server part is composed of signal control part and audio processing part, TTS and speech recognition part, VXML interpreter part. And, service scenario part is composed of service logic decision part, data access and storage part, and computation part. First server is composed of signaling part media processing part, which controls signaling and IVR functionality. Second server takes charge of service scenario and interpreting functions. Third server is used for database processing and TTS processing. Fig 4 shows more detailed architecture of ARS hosting service platform and its network topology.

Using VXML documents, separation of service logic part from IVR platform is accomplished. As a result of separation, it is possible for customers to edit their own web based service scenarios, service menus, ARS prompt.

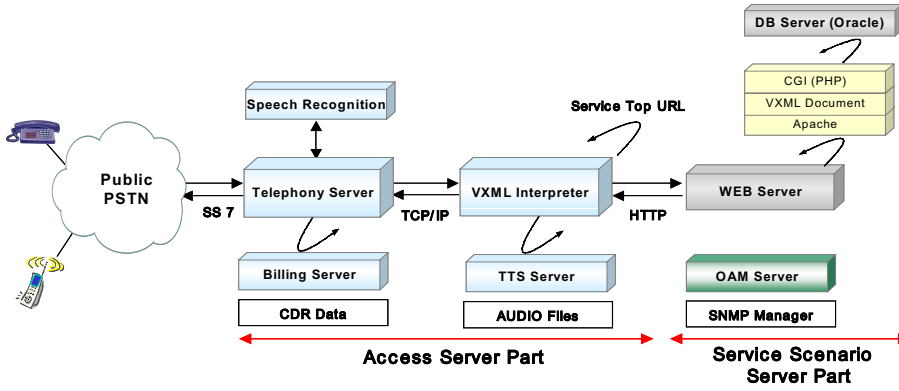


Fig. 4. Architecture of suggested CTI service platform and its network topology

3.1 Features of Suggested CTI Service Platform

We can summarize the features of suggested CTI service platform as follows.

First, we separated service scenario handling part from CTI platform. Second, our newly suggested CTI service platform provides web user interface to the customers, so that the customer can modify their own service scenario. This is the first implementation as commercial services. Third, our suggested CTI service platform controls signaling and audio resources, and provides speech enable functions. Fourth, we adopted VXML technologies to meet above suggestions.

4 Specification of Implemented ARS Hosting Service, Using Suggested CTI Service Platform

ARS hosting service can be characterized some features. Those who want to implement their own IVR service systems can use our CTI service platform, for the purpose of alternation. Primary customers would be company subscriber. If they use this ARS hosting service, there is no reason to implement their own service platforms.

It is a very cost effective method; company subscriber will not be requested additional cost of equipment investment. Also, they don't have to pay for the purpose of management of systems. Because, they only leased our (DACOM) IVR service platform, so they have no responsibilities to care systems.

The end user will contact specific IVR service systems with a universal access number (UAN). The end user can listen to prompt announcement or select special place to connect. Company subscribers, who want to serve IVR services to the end users, can offer their own services using our ARS hosting service platform. They can make their own service scenarios and change their service flows at right time easily,

using our web interfaces. This is a strong point of our newly suggested CTI service platform.

4.1 Implemented Service Overview

We assigned a new universal access number (UAN) to a company subscriber. Actually, we assigned 1544-xxxx. And then, company subscribers make their own service scenarios using our web page tools. Our web page provides GUI interface and service scenario editing tools. Subscribers can make their scenarios newly or change service flows easily. Actually, subscribers can contact their own homepage, change or edit their own service scenarios at http://1544ars.dacom.net. Finally, when the end users make a UAN calls, they are connected to the ARS center of subscriber. Fig 5 shows flows of implemented ARS hosting service.

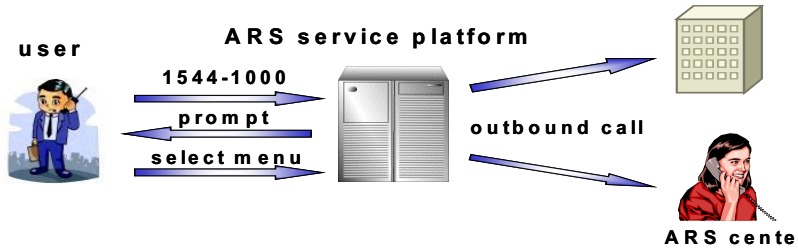


Fig. 5. Flows of implemented ARS hosting service

Our ARS hosting service has its own web pages. It provides very useful GUI functions to subscribers. Company subscribers can make up their own ARS service procedures, even if they don't know VXML script grammars. Creation of ARS service procedures using GUI wizard is very easy. If they use GUI wizard, VXML files are formed automatically, according to flows of their own service. Fig 6 shows implemented web page of our ARS hosting service.

Our ARS web page uses simple and efficient menu driven methods. Each menu allows voice prompt, DTMF detection and second call transfer functions. Especially, voice prompt can be used through various ways, including prerecorded audio files, recording by telephone, effect tone, and synthesized speech using TTS server. Company subscribers can modify their own service scenarios and change destined called numbers using Internet at any time. And then, company subscribers can confirm change of service flows by calling.

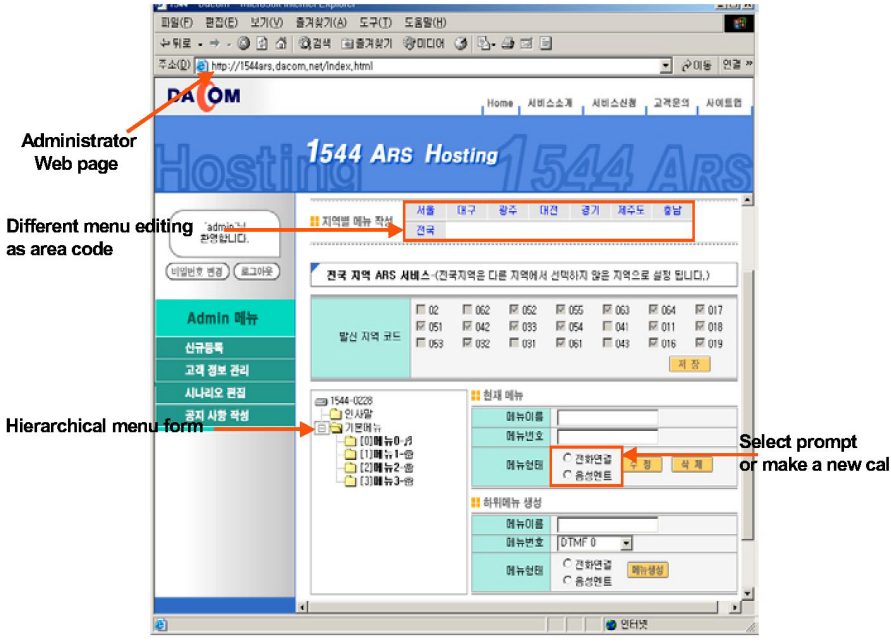


Fig. 6. Implemented web page of ARS hosting service

4.2 Specification of Implemented VXML Interpreter

VXML is a dialogue language defined by the VXML forum. VXML documents are extensible markup language documents, specifically designed to support voice services. Fig 7 shows architecture of implemented VXML interpreter.

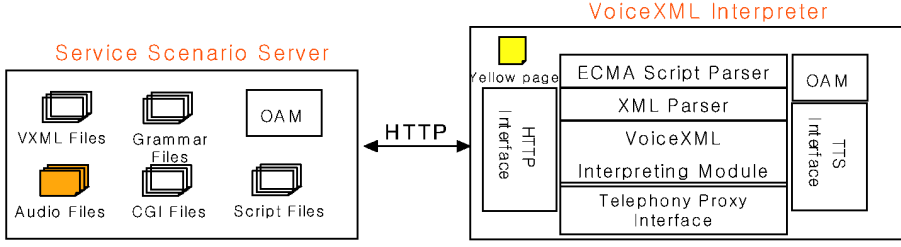


Fig. 7. Architecture of implemented VXML interpreter

Implemented VXML interpreter supports VXML 2.0 standard and adopts multi-threads technology. It also supports load-balancing and duplication structure. VXML interpreter also has watchdog processors for stability of systems.

4.3 Call Flows for ARS Hosting Service

Figure 8 shows simplified call flows of implemented ARS hosting service. Calling party calls service number, and then listen to ARS prompt. If he wants to connect a call center or other agent, it is possible to press DTMF buttons. If line of the call center is busy, telephony server would retry call process automatically.

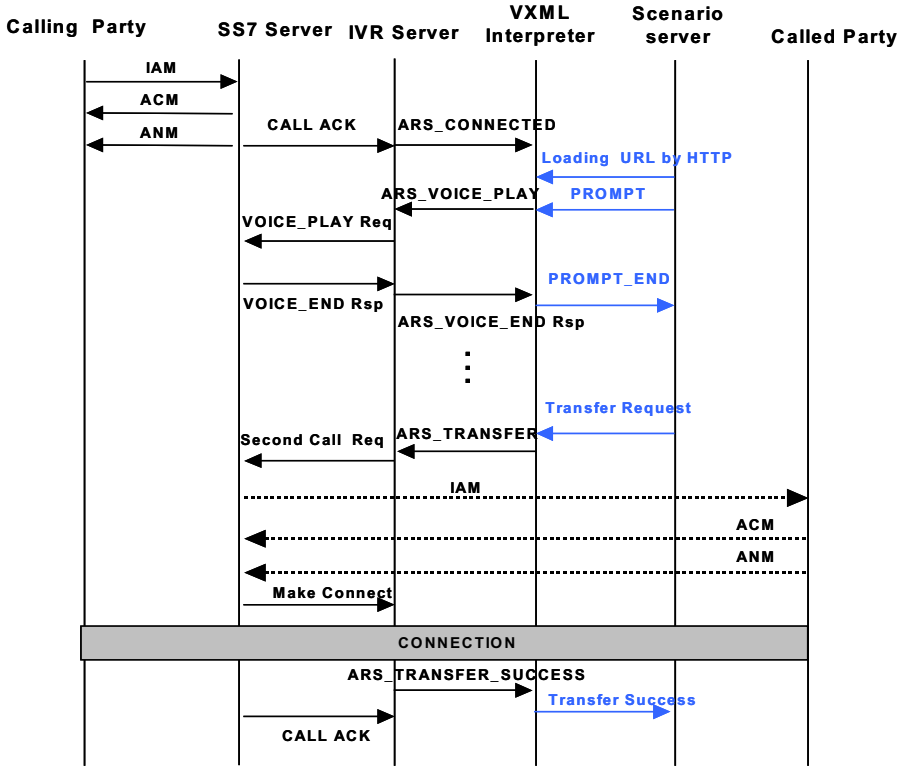


Fig. 8. Simplified call flows for implemented ARS hosting service

5 Analysis of Proposed New CTI Service Platform

We build a platform of new paradigm in commercial service. From beginning of commercial service to today, the subscriber's response is very affirmative. There is no need to know about VXML technologies. They can make their own service scenarios if they can use simple GUI tools. Principles of GUI tools are very easy. They can insert telephone numbers and upload prompt announcement files. They are pleased with accessibility of service platform using web user interfaces. There's no need to hire software engineers and operators for establishing their ARS services. These are

very cost effective methods to subscribers. Table 1 shows the results of analysis of proposed new CTI service platform.

6 Conclusions

The successful implementation of CTI service platform design for the subscribers depends on two factors: flexibility of service scenarios and possibility of change at proper time. We have shown that the efficiency of ARS hosting service based on VXML as company's solutions. And also, we have proved that the subscriber's response is very affirmative to our hosting services. Currently, over 200 customer subscribers are in served at our platform. Future work will involve extended applications such as voice web surfing and VAD (voice activated dialing) using excellent speech recognition systems. We were convinced that VXML solutions give chances to traditional PSTN carriers finding new killer applications.

Table. 1. Comparison between traditional IVR service platform and newly proposed platform

	Advantage	**Disadvantage**
Service architecture	-Open client-server model -Support VXML 1.0/2.0 standard -Efficiency of the internet infra utilization -Simplicity of service creation and modification -Flexibility of SSML	-Need multiple servers -Latency of Network
Customer convenience	-Maximum of service compatibility using Standard language -Efficiency of TTS and Voice Recognition -Dynamic assignment URL according to DNIS -Offer seamless service	
Development functionality	-Separation of IVR platform and service logic -Development of fast service scenario -Dynamic menu creation according to CID -Support of multiple application services -Interoperability of applications -Familiar with web developers	-VXML script support simple scenarios
OAM	-Enhancement of operation efficiency -Reduce maintenance costs	-Weak point. : instant bulk calls

References

1. Voice XML Forum: “VoiceXML 2.0 candidate Draft”: http://www.voicexml.org.
2. W3C proposed recommendation: “Speech Recognition Grammar Specification version 1.0”: http://www.w3.org/TR/speech-grammar .
3. T. Ball, V. Bonnewell, P. Danielsen, P. Mataga, K. Rehor: “Speech-Enabled Services Using TelePortal Software and VoiceXML” Bell Labs Technical Jounal July-September 2000.
4. DACOM R&D Center: “Telephony user interface specification at DACOM R&D Center” DACOM R&D Journal (2002).
5. Anonymousness: “ Telecom Cost Managements Services”: http://www.profitline.com.
6. Anonymousness: “ Telecom – Start Your Own Business”: http://www.telcan.net.
7. Bruce Lucas: “VoiceXML for Web-based Distributed Conversational Applications” Communications of the ACM, 2000
8. Quiane Ruiz, J.A.; Manjarrez Sanchez, J.R.: "Design of a VoiceXML Gateway", ENC'03, pp. 49~53, Sept. 2003.
9. Quiane Ruiz Jorge A.: "Design and Implementation of VoiceXML Gateway", Center for Computing Research-IPN, Mexico, 2003.

Storing Together the Structural Information of XML Documents in Relational Databases*

Min Jin and Byung-Joo Shin

Department of Computer Science and Engineering, Kyungnam University, Masan, KOREA
mjin@kyungnam.ac.kr, challenger@hawk.kyungnam.ac.kr

Abstract. There is a structural discrepancy between XML documents and relational databases. Hence, special data structures are required in order to represent the hierarchical structure of XML documents in storing the XML documents in relational databases. The structural information is required in querying XML documents and publishing the stored relational data in XML documents. In this paper, we propose a method for storing XML documents in relational databases, in which the structural information of XML DTD as well as the XML documents is stored. The structural information of XML documents is also stored. The XML DTD is restored by using the stored structural information and provided to users in creating queries against XML documents. XML documents can be also restored using the stored structural information. The XML queries written in XQuery are processed by exploiting the structural information of DTD that is represented in conjunction with the relational tables in which XML documents are stored.

1 Introduction

Due to the rapid development of the Internet technologies, XML is widely used as simple yet flexible means for exchanging data since it has nested, hierarchical, and self-describing structure. The volume of XML documents in Internet business environment is getting larger. This has given rise to the need for storing and querying increased volume of XML data. In general, XML data can be stored and queried in a file system, an object-oriented database system, an exclusive system for XML, or a relational database system. Relational databases are widely used in conventional data processing and provide inherent services such as transaction management and query optimization that can be exploited in managing XML data. Hence, a lot of studies have been focused on the use of relational databases to store and query XML data[5][6].

However, data are represented in flat structures in relational databases, whereas XML documents are represented in hierarchical structures with nested and recursive structures. Special data structures are required for storing and querying XML data in relational databases due to the structural discrepancy between XML and relational databases. Storing XML data with hierarchical structures in relational databases with flat structures gives rise to the following three issues. The first is that it is likely to

* This work was supported by Kyungnam University Research Fund.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 763–771, 2004.
© Springer-Verlag Berlin Heidelberg 2004

lead to excessive fragmentations and redundancy. The second is how to represent sub-elements that are sets and recursive elements of XML data in relational databases. The last is how to query XML documents stored in relational databases[12][16]. XML query languages such as XQuery[21] are not supported in relational databases. When users create queries in XML query languages, the structure of XML documents such as XML views in XPERANTO[2][9][11] and SilkRoute[4] should be provided to the users. The queries expressed in XML query languages are translated to the corresponding SQL statements. Otherwise, users create queries in SQL statements. The structural information of relational schema should be provided to the users in these schemes[3][10].

In this paper, we proposed a method in which not only XML documents but also the structural information of XML documents is stored in relational tables. The structural information can be retrieved by SQL. This information is used in restoring the hierarchical structure of XML documents, which is needed in creating XML queries. It is also used to restore the original XML document.

The rest of this paper is organized as follows. Section 2 briefly overviews related work concerning storing and querying XML data using relational databases. Section 3 describes our scheme for storing XML documents in relational databases. Section 4 describes how to store the structural information of XML documents together with XML documents in relational databases. Section 5 describes the implications of our method. Section 6 offers conclusions.

2 Related Work

Methods for storing XML documents in relational databases can roughly be classified into two categories: *Model Mapping Approach* and *Structure Mapping Approach*[17]. Model mapping approach deals with storing XML documents without structural information. Relational schemas are defined regardless of the structural information of an XML document. In this approach, an XML document can be represented as an ordered labeled directed graph, in which each element is represented as a node and relationships between element and subelement/attributes are represented as edges. Each node is labeled with a unique identifier and edge is labeled with the name of the sub-element. Values of XML documents are represented as terminal nodes. There are three alternative ways to store the edges of a graph; *Edge Approach*, *Binary Approach*, and *Universal Approach*. There are two alternative ways to store the values of a graph; a way to establish separate value tables for conceivable data types and the other way to inline values in the edge tables. Therefore, there are six ways to store XML documents in relational databases in the model mapping approach[5][17].

In contrast to model mapping approach, the structure mapping approach deals with storing XML documents with the structural information such as a DTD or a schema. Relational schemas are generated based on the structural information extracted from the DTD or the XML schema. Three techniques called *Basic Inlining*, *Shared Inlining*, and *Hybrid Inlining* have been proposed[12][17].

Intermediate systems such as XPERANTO, SilkRoute, and Rainbow have been proposed to cope with the structural discrepancy between relational databases and XML documents in querying XML documents stored in relational databases. A default XML view is provided to users creating queries in these systems. The user

XML queries are translated into corresponding SQL using the schema information[2][4][6][9][12][19].

In order to keep the order information of XML documents, numbering schemes have been proposed[8],[14],[18]. In the DFS numbering scheme[18], each object in the document is numbered in the sequence of depth first search. A variant of the original DFS numbering scheme was proposed, in which numbers are defined in the same sequence of DFS, there are gaps between numbers for accommodating insertions[7][18]. There is another method, which is called Bit Structured Schema[1] and it is also called the *Dewey Order Encoding* method[15]. The number of an object has the number of its parents and an additional number that is numbered according to the relative position among siblings. We can easily recognize the ancestor-descendant relationship between objects in this scheme. The work[15] provides the detailed explanation of the various order encoding methods and the performance.

3 Storing XML Documents in Relational Databases

In this paper, we adopt the structure-mapping approach for storing XML documents in relational databases. Our approach that is called the *Association Inlining* combines the join reduction properties of *Hybrid Inlining* with the sharing features of *Shared Inlining*[13]. A DTD is simplified and represented as a graph in Fig. 1.

```
<!ELEMENT Doc ( article*, paper* )>
<!ELEMENT article ( title, authors, editor )>
<!ELEMENT editor ( name )>
<!ELEMENT name ( #PCDATA )>
<!ELEMENT paper ( title, authors, reference? ) >
<!ATTLIST paper paperID ID #REQUIRED>
<!ELEMENT title ( #PCDATA )>
<!ELEMENT reference ( book*, paper* )>
<!ELEMENT book ( booktitle, authors )>
<!ELEMENT booktitle ( #PCDATA )>
<!ELEMENT authors ( author+ )>
<!ELEMENT author ( country?, university? )>
<!ATTLIST author name CDATA #REQUIRED>
<!ELEMENT country ( #PCDATA )>
<!ELEMENT university ( #PCDATA )>
```

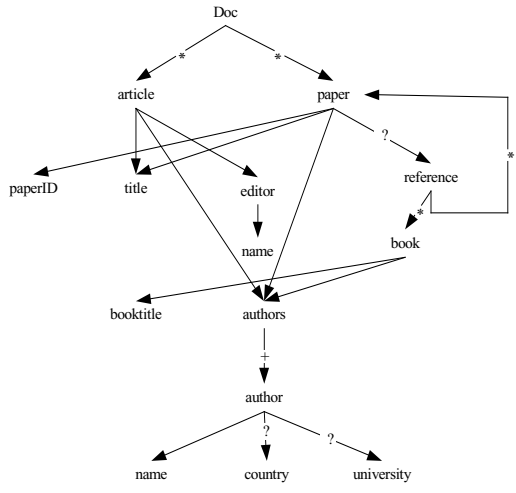


Fig. 1. A DTD specification and the DTD graph

Fig. 3 shows the created tables for the DTD graph in Fig. 1 and the corresponding XML document in Fig. 2 using the *association inlining*. Each relation has an *ID* field that serves as a key of the table. The *Author* table has a *parentID* field that serves as a foreign key that corresponds to its parent element node. The *parentCode* field points to the corresponding parent table among multiple parent tables. The *order* field

represents the occurrence order within the element. The *nested* field indicates the degree of recursions on the *parentCode* table.

```
<Doc>
  <article>
    <title>XML and RDBMS</title>
    <authors>
      <author name="Lee, D.J" />
    </authors>
    <editor>
      <name>Kim, Y.I</name>
    </eidtor>
  </article>
  <paper paperID="1">
    <title>Storing and Querying…</title>
    <authors>
      <author name="Shin, B.J">
        <country>Korea</country>
        <university>Kyungnam</university>
      </author>
      <author name="Jin, M">
            <country>Korea</country>
        <university>Kyungnam</university>
      </author>
    </authors>
    <reference>
      <book>
        <booktitle>Professional XML…</booktitle>
        <authors>
          <author name="Williams, K" />
        </authors>
      </book>
      <paper paperID="2">
        <title>Efficiently Publishing…</title>
        <authors>
          <author name="Shanmugasundaram, J" />
        </aurhos>
      </paper>
    </reference>
  </paper>
</Doc>
```

Fig. 2. An XML document

Article

articleID	parentID	parentCode	order	title	name	nestID	docID
1	NULL	NULL	1	XML and RDBMS	Kim, Y.I	NULL	1

Paper

paperID	parentID	parentCode	order	title	nestID	docID
1	NULL	NULL	1	Storing and Querying…	0	1
2	1	2	1	Efficiently Publishing…	1	1

Book

bookID	parentID	parentCode	order	booktitle	nestID	docID
1	1	1	1	Professional XML…	NULL	1

Author

authorID	parentID	parentCode	order	name	country	university	nestID	docID
1	1	1	1	Lee, D.J	NULL	NULL	NULL	1
2	1	2	1	Shin, B.J	Korea	Kyungnam	NULL	1
3	1	2	2	Jin, M	Korea	Kyungnam	NULL	1
4	1	3	1	Williams, K	NULL	NULL	NULL	1
5	2	2	1	Shanmugasundaram, J	NULL	NULL	NULL	1

Fig. 3. Tables for storing XML documents with DTD in Fig. 1

4 Storing Together the Structural Information of XML

4.1 Representing the Structure of XML DTD

The *DTD Structure Table* represents the hierarchical structure of the XML DTD. Here we use the *Dewey Order Encoding* for representing the ordering information among objects in the DTD since it could easily accommodate elements having in-degrees greater than one. We separate the DTD graph into several groups to cope with representing the order of elements having in-degrees greater than one. Each group has

an independent numbering scheme. The *DTD Structure Table* for representing the structural information of the XML DTD is given in Fig. 4.

ID	tag	groupID begin	groupID end	type	table	columnName	isRoot	edgeType	recursion
1	Doc	1	1	E	NULL	NULL	1	NULL	0
1.1	Article	1	1	E	article	NULL	0	*	0
1.1.1	Title	1	1	E	article	title	0	normal	0
1.1.2	Authors	1	2	E	NULL	NULL	0	normal	0
1.1.3	editor	1	1	E	NULL	NULL	0	normal	0
1.1.3.1	name	1	1	E	article	name	0	normal	0
1.2	paper	1	3	E	paper	NULL	0	*	0
1	authors	2	2	E	NULL	NULL	1	NULL	0
1.1	author	2	2	E	author	NULL	0	+	0
1.1.1	name	2	2	A	author	name	0	normal	0
1.1.2	country	2	2	E	author	country	0	?	0
1.1.3	university	2	2	E	author	university	0	?	0
1	paper	3	3	E	paper	NULL	1	NULL	0
1.1	paperID	3	3	A	paper	paperID	0	normal	0
1.2	title	3	3	E	paper	title	0	normal	0
1.3	authors	3	2	E	NULL	NULL	0	normal	0
1.4	reference	3	3	E	NULL	NULL	0	?	0
1.4.1	book	3	3	E	book	NULL	0	*	0
1.4.1.1	booktitle	3	3	E	book	booktitle	0	normal	0
1.4.1.2	authors	3	2	E	NULL	NULL	0	normal	0
1.4.2	paper	3	3	E	NULL	NULL	0	*	1

Fig. 4. The structural information of the XML DTD in Fig. 1

The *type* denotes the types of the node such as an element or an attribute. The *groupIDbegin* denotes the group from which the edge is incident and the *groupIDend* denotes the group to which the edge is incident. If *groupIDbegin* = *groupIDend*, the nodes connected by the edge are contained in the same group. Otherwise, they are contained in different groups. The *table* denotes a table that contains the corresponding information of the tag. The *columnName* denotes the column that corresponds to the tag in the table specified in *table* field. If *table* and *columnName* are NULL, the tag is not stored in the relational tables. If *table* has a value other than NULL and *columnName* is NULL, it means that the tag is mapped to a table name. In this case, the tag doesn't have any corresponding information in the relational database. The *edgeType* has one of the value in {+, ∗, ?, normal}. The *isRoot* represents whether the node is a root of the document or a group. If *groupIDbegin* = 1, then it is the root of the document. Otherwise, it is a root of a group. The *recursion* represents recursive relationships between nodes.

4.2 Representing the Structure of XML Documents

The structural information of XML documents is also stored in relational tables in our scheme. Here, the DFS numbering scheme is used since the XML document is different from the DTD. There are no nodes in the graph of an XML document having in-degrees greater than one. The structural information is used for rebuilding XML documents. The structure of an XML document is represented in the table as in Fig. 5.

Each tag in the XML document appears in the *tag* field. The *level* denotes the depth of the tag. The *begin:end* denotes the relative position of the tag in the document in

ID	tag	level	begin:end	table	columnName	tupleID	isRoot	Doc	DTD
1	Doc	0	0:65	NULL	NULL	NULL	1	1	1
2	article	1	1:15	article	NULL	NULL	1	1	1
3	title	2	2:4	article	title	1	1	1	1
4	authors	2	5:9	NULL	NULL	NULL	1	1	1
5	author	3	6:8	author	NULL	NULL	1	1	1
6	name	4	7:7	author	name	1	0	1	1
7	editor	2	10:14	NULL	NULL	NULL	1	1	1
8	name	3	11:13	article	name	1	1	1	1
9	paper	1	16:64	paper	NULL	NULL	1	1	1
10	paperID	2	17:17	paper	paperID	1	0	1	1
11	title	2	18:20	paper	title	1	1	1	1
12	authors	2	21:40	NULL	NULL	NULL	1	1	1
13	author	3	22:30	author	NULL	NULL	1	1	1
14	name	4	23:23	author	name	2	0	1	1
15	country	4	24:26	author	country	2	1	1	1
16	university	4	27:29	author	university	2	1	1	1
17	author	3	31:39	author	NULL	NULL	1	1	1
18	name	4	32:32	author	name	3	0	1	1
19	country	4	33:35	author	country	3	1	1	1
20	university	4	36:38	author	university	3	1	1	1
21	reference	2	41:63	NULL	NULL	NULL	1	1	1
22	book	3	42:51	book	NULL	NULL	1	1	1
23	booktitle	4	43:45	book	booktitle	1	1	1	1
24	authors	5	46:50	NULL	NULL	NULL	1	1	1
25	author	6	47:49	author	NULL	NULL	1	1	1
26	name	7	48:48	author	name	4	0	1	1
27	paper	3	52:62	paper	NULL	NULL	1	1	1
28	paperID	4	53:53	paper	paperID	2	0	1	1
29	title	4	54:56	paper	title	2	1	1	1
30	authors	5	57:61	NULL	NULL	NULL	1	1	1
31	author	6	58:60	author	NULL	NULL	1	1	1
32	name	7	59:59	author	name	5	0	1	1

Fig. 5. An XML document structure table

DFS order. If the *begin* is equal to the *end*, the tag means an attribute. The *table* denotes the table in which the corresponding information of the tag is stored. The *columnName* means the corresponding column of the table in which the value of the tag is stored. If *table* and *columnName* are NULL, the tag is not stored in the relational tables. If *table* has a value and *columnName* is NULL, it means that the tag is mapped to a table name. In this case, the tag doesn't have any corresponding information in the relational database. The *tupleId* denotes the identifier of the tuple that has the value of the tag in the relational table. The *isRoot* denotes whether an element appears as a column of a table that corresponds to one of the ascendant elements. The *doc* and *DTD* denote the identifier of the document and DTD respectively.

We can restore the original XML document using the stored structural information of the documents. The algorithm is given in Fig. 6.

5 Implications of Storing Together the Structural Information

Let us take a simple scenario to show the implications of our method. When a user creates a query against XML documents stored in relational tables, the corresponding XML DTD is restored from the stored structural information and provided to the user.

The algorithm for restoring XML DTD is given in Fig. 7. The original XML DTD is restored by exploiting the stored structural information which is shown in Fig. 4.

```
Algorithm RestoreDocument( XMLDataList ) {
/* Assume that all nodes in XMLDataList are tuples of a document. */
/* XD_cursor is the pointer to a tuple */
        index = 0; stack = NULL; resultString = NULL;
        XD_cursor = XMLDataList->FirstNode;
        While( XD_cursor != NULL ) {
                if( XD_cursor->begin == index && XD_cursor->end == index )
                        resultString += " " + XD_cursor->tag + "=" + XD_cursor->value;
                else if( XD_cursor->begin == index ) {
                        if( XD_cursor->begin != 0 )
                                resultString += ">";
                        resultString += "<" + XD_cursor->tag;
                        stack->push( XD_cursor );
                }
                else {
                        current_cursor = stack->pop();
                           if( current_cursor->end == index ) {
                                resultString += "</" + current_cursor->tag + ">";
                                break;
                        }
                        else
                                resultString += ">" + current_cursor->value + "</" + current_cursor->tag + ">";
                        index = current_cursor->end;
                }
                index++;
                XD_cursor = XD_cursor->NextNode;
        }
}
```

Fig. 6. Algorithm to restore the original XML document

```
Algorithm RestoreDTD( DTDStruct ) {
/* DTDStruct is a set of tuples of a DTD Structure table in Fig.4. */
/* Tuples are sorted by groupID_begin. */
/* Node is a structure that has the information of a tuple of the DTD structure table */
/* ChildNode is a structure of a child node of a node, which has an ID of the corresponding tuple and a pointer to a
sibling node */
        index = 0;
        DTDGraph[0] = ( Node_Pointer ) malloc( sizeof( Node ) );
        for( D_cursor = DTDList->FirstTuple; D_cursor != NULL; D_cursor = DTDList->NextTuple ) {
                DTDgraph[++index] = ( Node_Pointer ) malloc( sizeof( Node ) );
                if( D_cursor->groupID_begin == 1 && D_cursor->isroot == 1 )
                        parentID = 0;
                else if( D_cursor->isroot == 1 )
                        continue;
                else {
                        /* Search the parent element using groupID_begin and ID */
                        parentID = SearchParent(D_cursor->gropuID_begin, D_cursor->ID);
                }
                InsertChildNode( parentID );
        }
}
Algorithm InsertChildNode( parentID ) {
        ptr = DTDGraph[parentID];
        while( ptr->link != NULL )
                ptr = ptr->link;
        ptr->link = ( Node_Pointer ) malloc( sizeof( ChildNode ) );
}
```

Fig. 7. Algorithm to restore the original XML DTD

Then, a user creates a query written in an XML query language such as XQuery. Thus, the proposed method leads to the following consequences to be worthy of notion. First, the XML DTD is easily restored exploiting the structural information stored in relational tables whenever it is needed. The XML DTD is needed when users create queries in XML languages such as XQuery. If the restored XML DTD is not provided, something like a default XML view[2][5][6][9][11][19] should be provided to the user. However, it might be astray from the original XML DTD since a default XML view is generated from the relational tables without using the structural information systematically.

Second, it is consistent with the fact that XML documents are stored in relational tables. Thus, the structural information of DTD can be queried by using SQL. Third, the XML document can be easily restored. The structural information of XML documents is stored in relational tables in conjunction with tables for XML documents. Thus, the original XML documents are easily restored. Last, it leads to the consequence that queries on the hierarchical structure of XML documents are supported in the relational scheme. In other words, queries written in XML query language such as XQuery[21] can be processed via hierarchical SQL. Thus, our method leads to a novel method for processing XQuery expressions. We are working on this issue. The detailed explanation is out of scope of this paper.

6 Conclusion

In this paper, we have proposed a method for storing the structural information of XML documents together with XML documents in relational databases. The structural information of XML documents is stored in relational tables in conjunction with the tables for XML documents. We developed data structures for representing the structural information of both XML DTD and XML documents. The proposed method has some connotations in storing and querying XML documents. The original XML DTD can be easily restored whenever it is needed. It is necessary for a user to create queries. The original XML documents can also be restored very conveniently. The corresponding algorithms are also developed.

The method leads to an efficient way of translating XML queries written in XQuery into SQL queries and publishing desired XML documents. The method of processing XQuery queries via hierarchical SQL is being under development. The implications of our method for storing the structural information of XML documents together with XML documents in relational tables are to be evaluated against reasonable dataset.

References

1. Aguilera, V., Cluet, S., Veltri, P., Vodislav, D., Wattez, F.: Querying XML Documents in Xyleme. Proceedings of the ACM SIGIR 2000 Workshop on XML and Information Retrieval (2000)
2. Carey, D., Florescu, D., Ives, Z., Lu, Y., Shanmugasundaram, J., Shekita, E., Subramanion, S.: XPREANTO: Publishing Object-Relational Data as XML. Informal Proceedings of the International Workshop on the Web and Databases (2000) 105-110

3. David, M.M.: ANSI SQL Hierarchical Processing Can Fully Integrate Native XML. SIGMOD Record, Vol. 32, No. 1. (2003) 41-46
4. Fernandez, M., Kadiyska, Y., Morishima, A., Suciu, D., Tan, W.C.: SilkRoute: A Framework for Publishing Relational Data in XML. ACM Transactions on Database Systems (2002) 438-493
5. Florescu, D., Kossmann, D.: Storing and Querying XML Data Using an RDBMS. IEEE Data Engineering Bulletin, Vol. 22, No. 3. (1999) 27-34
6. Funderburk, J.E., Kiernan, G., Shanmugasundaram, J., Shekita, E., Wei, C.: XTABLES: Bridging Relational Technology and XML. IBM Systems Journal (2002) 616-641
7. Kha, D.D., Yoshikawa, M., Uemura, S.: An XML Indexing Structure with Relative Region Coordinate. Proceedings of 17th International Conference on Data Engineering (2001) 313-320
8. Li, Q., Moon, B.: Indexing and Querying XML Data for Regular Path Expressions. Proceedings of the 27th VLDB Conference (2001) 361-370
9. Shanmugasundaram, J., Kiernan, J., Shekita, E., Fan, C., Funderburk, J.: Querying XML Views of Relational Data. Proceedings of the 27th VLDB Conference (2001) 261-270
10. Shanmugasundaram, J., Shekita, E., Barr, R., Carey, M., Lindsay, B., Pirahesh, H., Reinwald, B.: Efficiently Publishing Relational Data as XML Documents. Proceedings of the 26th VLDB Conference (2000) 65-76
11. Shanmugasundaram, J., Shekita, E., Kiernan, J., Krishnamurthy, R., Viglas, E., Naughton, J., Tatarinov, I.: A General Technique for Querying XML Documents Using a Relational Database System. SIGMOD Record, Vol. 30, No. 3. (2001) 20-26
12. Shanmugasundaram, J., Tufte, K., He, G., Zhang, C., Dewitt, D., Naughton, J.: Relational Databases for Querying XML Documents: Limitations and Opportunities. Proceedings of the 25th VLDB Conference (1999) 302-314
13. Shin, B.J., Jin, M. Storing and Querying XML Documents Using a Path Table in Relational Databases. The 1st International Workshop on XML Schema and Data Management held in conjuction with ER2003 (2003) 285-296
14. Srivastava, D., Al-Khalifa, S., Jagadish, H.V., Koudas, N., Patel, J.M., Wu, Y.: Structural Joins: A Primitive for Efficient XML Query Pattern Matching. Proceedings of ICDE (2002) 141-152
15. Tatarinov, I., Viglas, S.D., Beyer, K., Shanmugasundaram, J., Shekita, E., Zhang, Chun. Storing and Querying Ordered XML Using a Relational Database System. SIGMOD Conference 2002 (2002) 204-215
16. Williams, M., Brundage, M., Dengler, P., Gabriel, J., Hoskinson, A., Kay, M., Maxwell, T., Ochoa, M., Papa, J., Vanmane, M.: Professional XML Databases. Wrox Press (2000)
17. Yoshikawa, M., Amagasa, T.: XRel: A Path-Based Approach to Storage and Retrieval of XML Documents Using Relational Databases. ACM Transactions on Internet Technology, Vol. 1, No. 1. (2001) 110-141
18. Zhang, C., Naughton, J.F., DeWitt, D., Luo, Q., Lohman, G.: On Supporting Containment Queries in Relational Database Management Systems. Proceedings of ACM SIGMOD Conference (2001)
19. Zhang, X., Pielech, B., Rundensteiner, E.A.: XML Algebra Optimization. Technical Report WPI-CS-TR-02-25, Worcester Polytechnic Institute (2002)
20. W3C Recommendation. XML Path Language (XPath) Version 1.0. In http://www.w3c.org/TR/xpath/ (1999)
21. W3C Recommendation. XQuery 1.0: An XML Query Language. In http://www.w3c.org/TR/xquery/ (2003)

Annotation Repositioning Methods in the XML Documents: Context-Based Approach[1]

Won-Sung Sohn[1], Myeong-Cheol Ko[2†], Hak-Keun Kim[1],
Soon-Bum Lim[3], and Yoon-Chul Choy[1]

[1] Department of Computer Science, Yonsei University,
Shinchon-dong, Seodaemun-ku, 120-749, Seoul, Korea
{sohnws, ycchoy}@rainbow.yonsei.ac.kr

[2]Department of Computer Science, Konkuk University,
Danwol-dong Chungju-si, Chungbuk, 380-701, Korea.
cheol@kku.ac.kr

[3] Department of Multimedia Science,
Sookmyung Women's University, 140-742, Seoul, Korea
sblim@sookmyung.ac.kr

Abstract. This paper presents context-based repositioning methods for annotations in the XML document. In the proposed methods, the XML-based original document and annotation information are presented as logical structure trees, and candidate anchors are produced in the process of creating matching relations between the trees. To select an appropriate candidate anchor among many candidates, repositioning rules are presented by stages based on the textual data and label information of anchor nodes of the logical structure trees. As a result, annotations in the structured document are robustly positioned even after various modifications of contexts in the document.

1 Introduction

The uses of annotation technique in the electronic document environment have rapidly expanded as they are more advantageous[1],[2],[3],[4] in the electronic document than in the paper document. Annotations in the electronic document are usually produced as fine grained and external link and saved within or outside of a system apart from the original document[2],[5],[6].

Therefore, they become orphan data when contents of the target document are deleted or modified[1],[2],[6],[7]. To solve that problem, annotations' anchor repositioning (anchoring) should be possible even after modifications of the original document[1],[6]. The repositioning process is considered as the most important function of an annotation system[3].

[1] This work was supported by the Post-doctoral Fellowship Program of Korea Science & Engineering Foundation (KOSEF)

† Contact Author

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 772–782, 2004.
© Springer-Verlag Berlin Heidelberg 2004

The annotation repositioning process involves the relations between annotations and sub-resources including contents and spans presented as external links. Therefore, the contexts between annotated information and the document should be considered in the annotation's anchor repositioning[2],[6]. Actually most robust annotation repositioning methods primarily consider contexts like unique IDs, substrings, surrounding texts, and keywords, as shown in table 1.

However, related works on the annotation repositioning detects only moves of texts in the original document[4],[8],[9],[10]. Therefore, if texts are modified, all the related annotations are usually orphaned[1],[6].

The robust anchoring method is asked for in particular in the XML-based annotation environment[5] because XML is the original document in the programs that use annotations the most frequently such as Cyber-Class, e-Learning, and e-Book[13]. However, most of the related works deal with only text documents, and other works that consider structured documents[6],[7] determine modifications of the document by comparing only the paths between annotations and the original document. Therefore, the repositioning of annotations is difficult when structures of the original document were deleted, moved, or modified.

Table 1. Comparison of annotation repositioning methods.

Approach Types	Repositioning Methods	Characteristics	Limitations
Unique Context [4],[8]	Edited or not determined by the existence of the unique context (id, substring)	Well applicable to various systems	Annotations are usually orphaned or deleted if unique substring detection is failed.
Redundant Context [9],[10]	Edited or not determined by comparing the anchor texts and original documents, and selects final positions by comparing surrounding texts.	Applied in most annotation systems that provide anchor repositioning	Does not consider selection of candidate anchors that are created in the process of repositioning on modified texts.
Keyword Anchoring [1],[2]	Extracts unique keywords from anchor texts, and detects anchor positions based on the keywords.	Reflects the cognitive feature for anchor detection and provide robust anchoring interface that uses confidence scores of anchors	Handles only general text documents, and it cannot operate without keyword-finding processes even in a huge-sized web environment.
Tree Walks [6],[7]	Checks out unique identifier between original documents and anchors, and then selects proper nodes through tree walks if updates are found.	Attempt repositioning even after documents were modified. And, it provides interface through which new anchors are reattached if anchoring is failed.	Tree walk method performs only path matching operations between HTML documents and annotations.

This paper presents context-based annotation repositioning methods in the XML-based annotation system. The proposed repositioning methods present the XML origi-

nal document and annotation information as logical structure trees and create matching relations between the trees. Candidate anchors are created in the process, and repositioning rules are presented by stages for selecting an appropriate anchor among the candidates. The proposed repositioning rules are about creating and merging candidate anchors based on the label and textual contexts of anchor nodes of the logical structure trees.

In that way, annotations are robustly repositioned even after contexts are deleted, moved, or modified in the structured document.

2 Context-Based Repositioning of Annotations

This paper proposes annotation processing methods for its robust repositioning in the XML document. In the proposed methods, logical structure trees are created for the target document (XML) and annotations, and annotations are robustly repositioned in the modified original document, as shown in Figure 1.

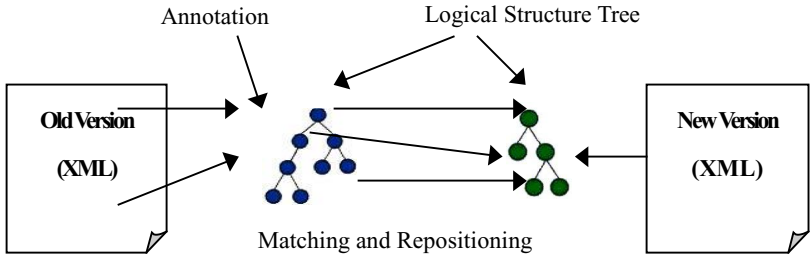


Fig. 1. Overall processing of the proposed annotation repositioning methods.

To find out how much the target document was modified, the proposed methods determine whether created annotation information and document structures differ with each other by traversing the logical structure trees. According to the degree of the modifications, matching relations[12] are created between the nodes of the trees, and proper candidate anchors are created by stages in the process. To obtain results that can be properly applied in the annotation environment, this work considered various changes of paths in the logical structure trees and anchor texts. The proposed repositioning methods are divided into robust repositioning in the document where structures remain the same and robust repositioning between documents of different structure information. The details of the proposed methods are as follows.

2.1 Annotation Repositioning between the Non-changed Structures

The proposed methods follow different repositioning processes depending on whether structures of the original document were modified or not. To do that, the methods examine how annotation anchors' path, offset, and anchor text information exist in the original document by traversing the logical structure trees of the original document first. If annotations' path information remains the same in the original document, only whether there were modifications between anchor texts is determined. In this work, the following comparing function – the longest common subsequence rate[13], which is based on the longest common subsequence[12],[13], was used for determining whether there were modifications between annotations' anchor texts and the original document's text nodes.

$$LCSR(x, y) = \frac{2\times|\,lcs(x, y)\,|}{|\,x\,|+|\,y\,|} \quad (1)^2$$

If the structures between annotation paths and the original document were not changed, the following repositioning rule 1 and 2 are applied to extract proper anchors. The details are as follows.

Repositioning Rule 1: *Let's suppose that there exist annotations' anchor text nodes,* $T_1 = \{x_i\}$, 1 *i s, and the original document's text nodes* $T_2 = \{y_i\}$, 1 *i* , *on annotation information* T_1 *and annotations' target document* T_2 *respectively. Each text node includes textual data* $x_i = \{T_1strin\}$, 1 *n* / *and* $y_i = \{T_2strin\}$, 1 *n* , *and each textual data includes characters* $T_1str_{ik} = \{a_{ik}\}$, 1 *n* *k and* $T_2str_{ik} = \{b_{i?}\}$, 1 *n* 0. *Create matching relations,* $[x_{io}, y_{iq}],\ldots,[x_{ip}, y_{ir}]$ *if all the nodes of annotation information* T_1 *exist in* T_2 *in the same label and order, and also there exist anchor text nodes* $x_{io},\ldots, x_{ip}$ *of* T_1 *and text nodes* $y_{iq},\ldots,y_{ir}$ *of* T_2 *having the same parent node labels each other.*

If the detection rule 1 is satisfied, it can be assumed that the original document's paths were not changed. In that case, one-to-one matching relations between the anchor texts can be created. For instance, as shown in Figure 2, if the original document's structures were not changed, one-to-one matching relations between paths of annotations and the original document can be created as [25, 25] and ①, [27, 27] and ②, [28, 28] and ③, and [36, 36] and ④.

If matching relations were created by the repositioning rule 1, similarity rates between anchor texts should be measured. Annotations are repositioned according to the measuring results. The details are explained in the repositioning rule 2.

Repositioning Rule 2: *If the LCSR between the nodes under the matchings created in the rule* 1 *is* 1, *the original document's text nodes* $y_{iq},\ldots,y_{ir}$ *are designated as anchoring boundaries. If the LCSR is between a threshold value and* 1, *text nodes* $y_{iq},\ldots,y_{ir}$

[2] |lcs(x,y)|, |x|, and |y| denote length of LCS between text x and y, length of x, and length of y, respectively.

are regarded as candidate anchors. If the LCSR is below a threshold value, which means anchor information of the old version were deleted or structures were modified, repositioning rule 3-6 *are applied again. If there is no text node in* T_2 *that satisfies rule* 3-6, y_{iq} *is appointed as a candidate anchor.*

After the repositioning rule 1 and 2 are applied, if it is found that anchor texts remain the same or were updated without changes in paths, either the old boundaries are used, or new candidate anchor boundaries are extracted. For instance, as the LCSR of the matching [25, 25] and ① in Figure 3 is 1, the node [25] is anchored right away. The LCSR of the matchings [27, 27] and ②, and [28, 28] and ③ are between a threshold value and 1. Therefore, nodes [27] and [28] of T_2 are appointed as candidate anchor boundaries. On the other hand, the similarity rate of the matching [36, 36] and ④ is below a threshold value. Therefore, a new repositioning rule should be applied.

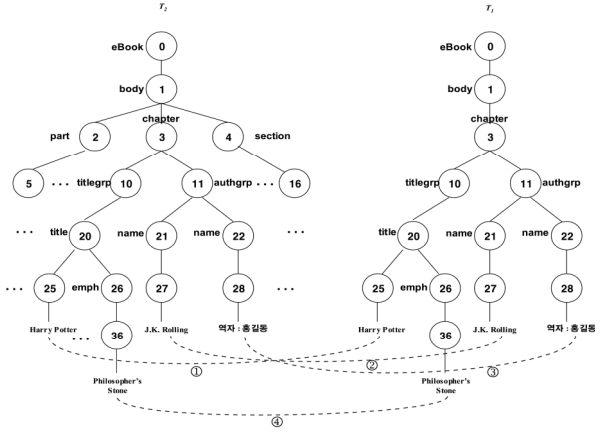


Fig. 2. When relations between annotation paths and the original document were not changed.

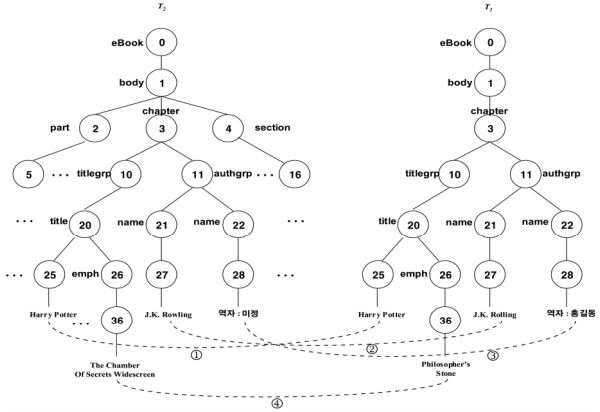


Fig. 3. An example of repositioning between anchor nodes of the same path.

2.2 Annotation Repositioning between the Changed Structures

If there were modifications in paths and anchor texts between the original document and annotations, similarities between the structures should be considered. Moreover, candidate anchors selected by the information include one-to-one, one-to-many, many-to-many, and many-to-one anchors on anchor texts and paths, due to the special features of annotations. Accordingly, to select proper candidate anchors, this work considers path matching methods by stages, and candidate anchor merging and linking methods. The details are as follows.

Repositioning Rule 3: *Let's suppose that path information of T_1, which is annotation information, does not coincide with node labels and sibling orders of T_2, the annotations' target document. Extract the LCSR between $x_{io},\ldots,x_{ip}$, annotation anchor text nodes of T_1, and all the text nodes of T_2, select text nodes of T_2 above a threshold value, and accordingly create matchings $[x_{io}, y_{iq}],\ldots,[x_{io}, y_{ir}],\ldots,[x_{ip}, y_{iu}],\ldots,[x_{ip}, y_{iv}]$.*

If the original document's structures and contents of text nodes were modified, repositioning rule 3 should be applied first, for creating matching relations where the LCSR are above a certain value. For instance, as shown in Figure 4, for node 25, matchings [25, 70] and ①, and [25, 58] and ② can be created, and matchings ③, ④, ⑤, ⑥, and ⑦ can be created according to the same rule.

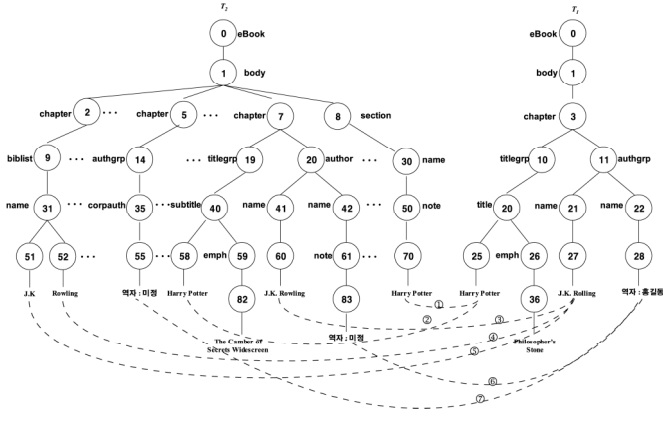


Fig. 4. Creation of candidate anchors based on the text LCSR.

Matchings created between text nodes can be many anchor nodes to many text nodes, as shown in Figure 4. In that case, the proposed methods create new matchings related in meanings by comparing the LCSR between node labels, rather than selecting one matching relation merely based on the LCSR between text nodes.

Repositioning Rule 4: *If there exist many matchings where the LCSR of the original document on anchor nodes are above a certain value, compare the label LCSR be-*

tween paths[3] *of matched text nodes, and then create matchings* $[x_{io}, y_{iq}],\ldots,[x_{io}, y_{ir}],\ldots,[x_{ip}, y_{iu}],\ldots,[x_{ip}, y_{iv}]$ *that include label similarity rates above a certain value.*

According to the repositioning rule 4, among many matchings based on text similarity rates, the matchings in which the label similarity rates are above a certain value, are appointed as new candidate anchors. For instance, as shown in Figure 5, in matchings in black lines, [25, 58] and ②, [27, 60] and ③, [27, 52] and ④, [27, 51] and ⑥, and [28, 83] and ⑦, the label LCSR between paths are above a certain value, therefore they are appointed as new matchings. Likewise, if there exist many candidate anchors that satisfy both rule 3 and 4, they are regarded as having higher priority and used as the rule for anchor recommendation interaction later in the repositioning interface.

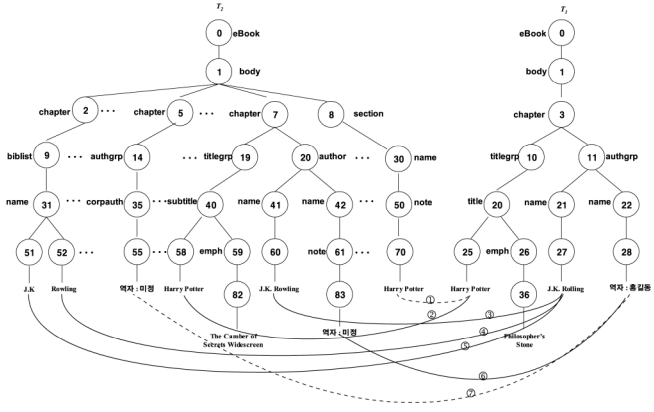


Fig. 5. Creation of candidate anchors based on the label LCSR.

If there are matchings that have the same label similarity rate among those created by the rule 4, the possibility of merging the matchings should be considered by using similarity rate information and the adjacency between labels, for reducing the possible number of matchings. Details are explained in the rule 5.

Repositioning Rule 5: *If there exist many nodes of which label similarity rates between paths are above a certain value after the rule* 4 *is applied to text nodes* $y_{iq},\ldots,y_{ir}$ *of the original document, determine whether the nodes can be combined each other. For that purpose, merge sibling text nodes that have a same parent node or text nodes that have a series of orders, and regard them as candidate anchor boundaries.*

Among the matchings created by the rule 4, those that satisfy the merging rule 5 should be merged as anchor boundaries, and selected as new candidate anchors. For instance, as shown in Figure 6, the matchings [27, 51] and [27, 52] are in sibling

[3] Path (x) means a sequential set of nodes from a parent node to a root node of node x.

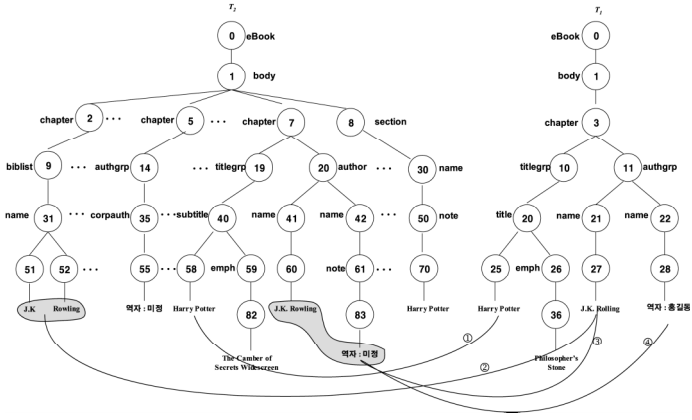


Fig. 6. Creation of candidate anchors by merging nodes

relations that have a same parent node, and text nodes of matchings [27, 60] and [28, 83] have a series of orders in the original document. Therefore, they are merged as anchor boundaries and used later as references.

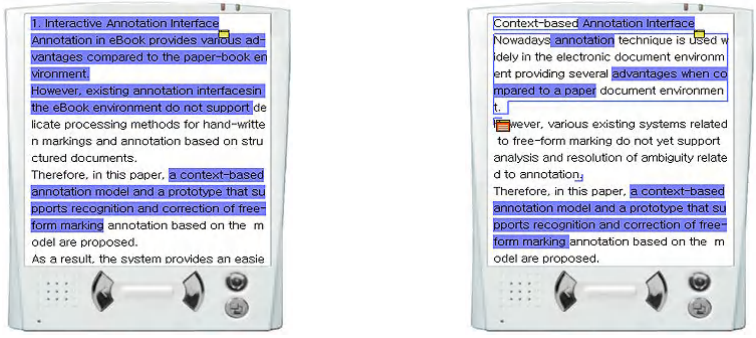
3 Implementation Result

In this section, the results of implementing the annotation system including the proposed methods and interface are examined. The system uses the XML-based eBook standard[11], and operates in the window XP and CE environment. In this work, the annotation system operated in the window CE.

Figure 7(A) shows that robust anchoring is provided by applying the proposed methods and interface to target documents. Figure 7(B) shows primitive anchoring on annotations ①, ②, and ⑤ in Figure 7(A) displayed as annotations in the rectangle form including in particular common anchor texts extracted by LCSR as highlights, which is to provide users with the anchor selection criteria. At the annotation ③ in Figure 7(A), contents of the anchor texts were severely modified, although the anchor structures remain the same in Figure 7(B).

4 Experimental Evaluation

For the purpose of evaluating the efficiency of the proposed methods, empirical user tests were conducted. This experiment deployed prototypes that applied the proposed system, the Robust Location method[6] that uses structure information, and the WebVise[9] method that uses surrounding texts and anchors' offset information instead of structure information.



(A) (B)

Fig. 7. Original document where annotations were inserted(A) and anchoring results(B).

Users created 50 annotations in the XML-based e-Book document, using three prototypes. And each prototype repositioned 20 annotations of which structures and anchors had been changed, 20 annotations of which only anchor texts had been changed, and 10 annotations where nothing had been changed after the original document had been changed. And then users were asked to fill out question sheets that used the scales of the lowest accuracy 1 and the highest 10 for evaluating the prototypes' repositioning accuracy.

A Single-Factor ANOVA (analysis of variance) was used to analyze performances by subjective accuracy. Figure 8 shows subjective ratings for the accuracy of repositioning of annotations according to application of each method in evaluation. Significant main effects were seen between the three results of repositioning with regard to application of each of the methods ($F(2,57) = 8.98$, $P < 0.05$).

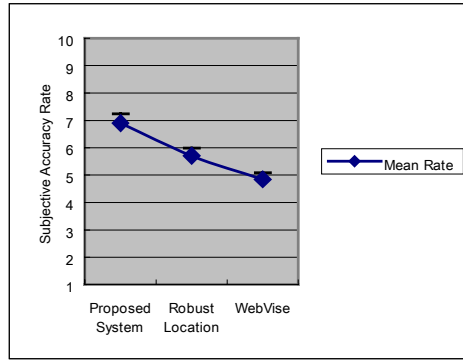


Fig. 8. Subjective evaluation of the accuracy by applying each method in Experiment 1 (1 = lowest accuracy, 10 = highest accuracy).

The results showed that the method that obtained the highest accuracy rate in the structured document was the one proposed in this paper. In particular, it was found out that the proposed candidate anchor creation method by stages influenced the users' evaluation of the accuracy more effectively than the Robust Locations' tree walk

method that used structure information. On the other hand, the method that used only surrounding texts disregarding structure information showed lower satisfaction rates compared to the proposed method. It seems that repositioning results in the structured document are affected by the use of meaningful element relations like path information as well as similarity rates of textual data.

5 Conclusions and Future Works

This paper proposed robust repositioning methods and systems for annotations in the XML document. In the proposed methods, the XML original document and annotation information were presented as logical structural trees, and matching relations were created between the trees. Repositioning rules were presented by stages for selecting an appropriate anchor among many candidates.

The proposed repositioning rules used the similarity rate on anchor text information to detect the first anchoring point, and the similarity rate on paths to create candidate anchors and extract meaningful information from the structure information. The boundary of candidate anchors was effectively determined by the merging rules. Finally, the created candidate anchors were either recommended and selected, or orphaned through the user interaction.

In that way, annotations in the structured document were robustly repositioned even after contexts of the document were deleted, moved, or modified. This work can be applied in annotation systems that use structured documents such as XML/SGML as well as in the web (HTML) and can be effectively applied to the online text editing, eBook, Cyber-Class, and so on. More work will be conducted in the future on applying the methods to the annotation interface that includes semantic information creation and robust anchoring operations in the XML-based semantic web environment.

References

1. Bernheim, A.J., Brush & Bargeron, D. (2001). Robustly Anchoring Annotations Using Keywords. Technical Report, MSR-TR-2001-107, Microsoft Research
2. Brush, A.J., David, B., Anoop, G. & Cadiz, JJ. (2001). Robust Annotation Positioning in Digital Documents. Proceedings of CHI'01. Seattle, March 31, ACM Press, NY, 285-292.
3. Cadiz, J., Gupta, A., & Grudin, J. (2000). Using Web Annotations for Asynchronous Collaboration Around Documents. Proceedings of CSCW '00. Philadelphia, ACM Press, NY, 309-318.
4. Ovsiannikov, I. A., Arbib, M. A. & Mcheill, T. H. (1999). Annotation Technology. International Journal of Human-Computer Studies. 50 (4), 329-362.
5. Davis, H. C. (2000). Referential Integrity of Links in Open Hypermedia Systems. Proceedings of ACM Hypertext '98. Pittsburgh, ACM Press, NY, 207-216.

6. Phelps, T. A. & Wilensky, R. (2000a), Robust Intra-document Locations. Proceedings of the 9th WWW Conference. Amsterdam.
7. phelps, T. A. & Wilensky, R. (2000b). Multivalent Documents. Communications of the ACM. 43 (6), 83-90.
8. Roscheisen, M. & Winograd, T. (1995). Shared Web Annotations as a Platform for third-party Value-added Information ProvidersTechnical Report, CSDTR/DLTR, Stanford University.
9. Grønbæk, K., Sloth, L. & P. Ørbæk (1999). Webvise: Browser and Proxy Support for Open Hypermedia Structuring Mechanisms on the WWW. Proceedings of the Eighth World Wide Web Conference. Toronto, Canada.
10. Yee, K.-P. (1997). The CritLink Mediator. http://crit.org/critlink.html.
11. Sohn, W. S., et al., (2002). Standardization of eBook documents in the Korean Industry. Computer Standards & Interfaces. 24(1), 45-60.
12. Chang, G. J. S., Patel, G., Relihan, L. & Wang, J. T. J. (1997). A Graphical Environment for Change Detection in Structured Documents. Proceedings of Twenty-First Annual Int'l Computer Software and Applications Conference (COMPSAC'97). Los Alamitos, CA, 536-541.
13. Lee, K. H., Choy, Y. C., & Koh, K. (2001). Change Detection of Structured Documents using Path-Matching Algorithms. Journal of KISS(Korean). 28 (4).

Isolating and Specifying the Relevant Information of an Organizational Model: A Process Oriented Towards Information System Generation[1]

Alicia Martínez[1,2], Oscar Pastor[1], and Hugo Estrada[1,3]

[1] Technical University of Valencia
Avenida de los Naranjos s/n, Valencia, Spain
{alimartin, opastor, hestrada}@dsic.upv.es
[2] I.T. Zacatepec, Morelos, Mexico
[3]CENIDET Cuernavaca, Mor. Mexico

Abstract. Recently, many research efforts in software engineering have been focused on integrating the organizational modeling in the software production process. The objective of these approaches is to use the organizational needs as the starting point for the generation of the information system, allowing us to assure that the functionality of the information system adequately corresponds to the tasks that are executed in the business. However, one of the principal problems of the current research work in this field is the lack of a methodological approach to isolate the relevant information to be automated by the information system. This lack is a real obstacle to producing specification of information systems from organizational models, due to the non-existence of a mechanism to filter out the irrelevant tasks. In this paper, a methodological approach for isolating and formally specifying the relevant information of an organizational model represented in Tropos Framework is presented. By doing this, we go a step further in the process of including business modeling as a key piece in the software production process.

1 Introduction

At present, many research works in software engineering have been done to assure to the correct construction of the software product; various techniques, methodologies and tools to achieve the objective of correctly implementing an information systems have been proposed [1],[2],[3]. However, only a few research efforts are focused on the problem of reducing the real impedance mismatch between the software product and its operational environment. This non-correspondence, makes it impossible for the information system to have the necessary functionality to permit the organizational actors to perform their organizational tasks.

One of the principal problems of the current research works in this field is the lack of a methodological approach to isolate the relevant information to be automated by

[1] This work has been partially supported by the MCYT project with ref. TIC2001-3530-C02-01, the Technical University of Valencia, Spain and the National Association of Universities and Institutions of Higher Education ANUIES, Mexico.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 783–790, 2004.
© Springer-Verlag Berlin Heidelberg 2004

the information system. The lack of a mechanism to filter out the irrelevant task is a real obstacle to producing an information system from an organizational model, due to the fact that the organizational model contains a lot of information which does not need to be taken into account in the definition of the expected functionality of the information system.

Another important issue in organizational modeling is the definition of a formal language to represent the modeling elements. The creation of this specification is essential for analyzing the improvement of the model and also for considering the automatic generation of specifications of the information system from the organizational model. However, there are only a few research efforts in this field.

In this paper, we propose a methodological approach to generate the formal specification of the relevant information of an organizational model. To do this, we propose using the Tropos Framework [4], a well-known technique, to represent the organizational model. Then the organizational model is specified using a formal language based in KAOS [5].

The paper is structured as follows: Section 2 presents the Tropos Framework. Section 3 presents an overview of the proposed method. Section 4 presents the selection of the relevant information. Section 5 shows the specification of the organizational model. Finally, section 6 presents the conclusions and further work.

2 The Tropos Methodology

Tropos [4] proposes a software development methodology and a development framework, which is based on the premise that in order to build software that operates within a dynamic environment, it is necessary to analyze and explicitly model that environment in terms of actors, their goals and their dependencies on other actors.

To support modeling and analysis during the early requirements, Tropos adopts the concepts offered by i* [6]. In Tropos, we have the following concepts: Actors, Goal Dependencies, Task Dependencies, Resource Dependencies and Softgoal dependencies. To details of the Tropos notation see [4].

By using these elements, it is possible to create the i* Strategic Models: the Strategic Dependency Model and the Strategic Rationale Model.

The *Strategic Dependency Model* (SD) shows the dependencies that exist between actors in a organizational process.

The *Strategic Rationale Model* (SR) shows the tasks that have to be carried out by the actors to achieve their goals and fulfill their dependencies.

The Tropos methodology has been used in several application areas, including requirements engineering, software processes and business process reengineering.

3 Outline of the Methodology Proposed

In this section, we present a general overview of the two phases that compose the proposed method.

Phase 1. Use the Tropos Framework to select the relevant information from the organizational model. This phase produces an organizational model that integrates the

software system actor. This system actor contains the relevant organizational goals to be automated by the software system.

Phase 2. The organization model is specified using an extension of the KAOS formal language [5]. Each enrichment that is introduced to the KAOS language will be pointed out.

In order to illustrate our approach, we used the *Paper Review Process* case study. The purpose is to model the process of reviewing of papers for a conference. In the case study, the Authors send the papers to the President of the Program Committee (PcChair), who selects the members of program committee (PcMember). The PcMembers are responsible for evaluating the papers and for sending the evaluations to the PcChair indicating acceptance or rejection. Finally, the PcChair send the notifications to the Authors. As a final result of the modeling process, we expect to obtain a software system that handles the process of submission, assignment, evaluation and selection of papers for the conference.

4 Selecting the Relevant Information of the Organizational Model

The first step of this process is the selection of an organizational model represented in the Tropos Framework. The next step of this phase consists of inserting the software system actor in the organizational model with the objective of determining the type of interaction of each organizational actor with the software system actor. An important concept used in this process is the concept of "module". A module represents the set of tasks performed by the actor to satisfy a goal. Too often, an actor has more than one goal in the organization; hence, the actor has more than one *module*.

The next step consists of using the soft goals to identify the modules that need to be delegated to the system actor. These modules need to be moved to the software system actor, additionally, the name of the actor that was the container of the original module must be included in the module. Finally, it is necessary to create new resources dependencies and task dependencies to allow the actors to satisfy their dependencies.

The process for constructing the organizational model and of including the software system actor is out of the scope of this paper. However, this process is presented in depth in [7].

As a result of the process of selection of relevant information, a new organizational model is created. For the sake of brevity, in this paper, we show only a small fragment of the Strategic Rationale Model for the case study (Figure 1). In this model, the software system actor depends on the Author to submit a paper (specifying the concrete way to perform this task) and to obtain the resource Paper. The Author, in turn, depends on the system actor to obtain the notification of the paper.

The PcChair depends on the software system actor to assign the papers to the PcMembers and to send the notifications to the Authors. In turn, the software system actor depends on the PcChair to obtain the list of the PcMembers. In this way, the system actor has the modules*: assign the paper to PcMembers*, *send notifications to the Authors* and *obtain papers*.

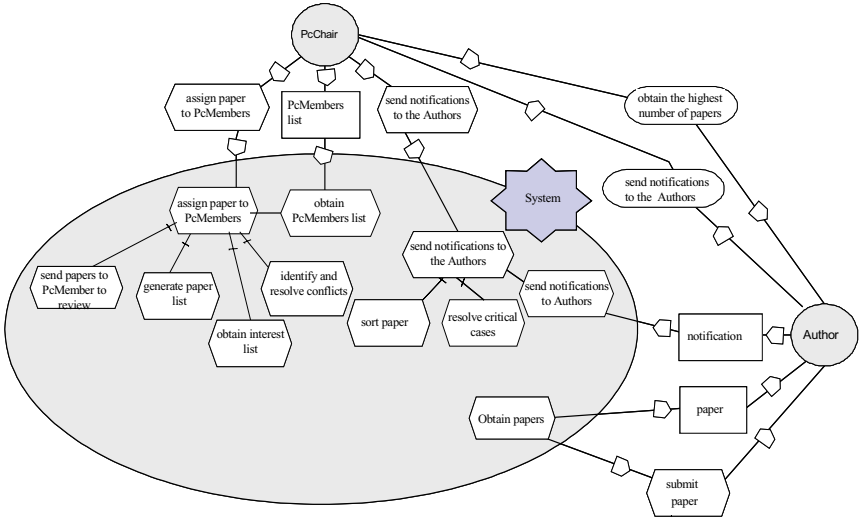


Fig. 1. A fragment of the Strategic Rationale Model for the case study *Paper Review Process*

5 Creating the Specifications of an Organizational Model

Starting from the original KAOS proposal [5], in this paper, we add a few elements to it to specify the particularities of a Tropos Organizational Model in the scope of our work[2]. In consequence, we propose to use the KAOS-like specification language to represent the elements of the new organizational model. KAOS [5] is a formal framework based on temporary logic to elicit and represent the goals that the system software should achieve.

In spite of the fact that KAOS was developed to represent information system goals, its syntax and semantics turn out to be especially appropriate for formally representing the conceptual primitives of the strategic models of the Tropos framework. This is due to the capacity of KAOS to represent modeling concepts at diverse abstraction levels.

In the following section, we define a set of specification rules to specify each one of the elements of the Tropos organizational model in the KAOS-like language.

5.1 Actor Specification

The actors of the organizational model are represented using the concept of *Agent* of the KAOS language. An agent is an object which is a processor for some actions [5]. The definition of an agent is composed of the agent name as well as a list of attributes.

In the fragment of the case study analyzed, there are two actors that have dependency relations with the software system actor: *PcChair* and *Author*. Table 1 shows the specification of the *Author* actor.

[2] In this section, each enrichment that is introduced will be pointed out.

Table 1. Specification of the actor *Author*

Agent Author Has name, email, affiliation: String End Author

5.2 Goal Specification

The goal dependencies of Tropos are represented using the concept of *SystemGoal* of the KAOS language. *SystemGoals* are application-specific goals that must be achieved by the composite system [5]. The definition of the *SystemGoal* is composed of:

a) The type of the goal (*Achieve, Cease, Maintain, Prevent and Optimize*[5]). In our proposal, we use *Achieve Goals* to express the fact that the goal needs to be satisfied in a current or a future state.
b) The specific category of the domain-level goal (*SatisfactionGoal, InformationGoal, SafetyGoal, ConsistencyGoal and RobustnessGoal*) declared in the clause *IntanceOf*. In this case, we use *SatisfactionGoals* to express the satisfaction of the agent request.
c) The clause *Concerns,* which links the goal with the objects. In this case, we use the clause *Concerns* to indicate the *depender* and *dependee* actors.

Table 2 shows the specification of the goal "*obtain the highest number of papers*".

Table 2. Specification of the goal *Obtain the highest number of quality papers*

SyemGoal Achieve [Obtain the highest number of papers] InstanceOf SatisfactionGoal Concerns PcChair, Author

5.3 Resource Dependency Specification

The resource dependencies of the organizational model are represented using the concept of *Entity* of the KAOS language. An *Entity* in KAOS is an autonomous object; its instances may exist independently from the other instances [5]. The *Entity* definition is composed of the entity name as well as a list of attributes.

In our case study, there are three resource dependencies between the organizational actors and the software system actor: *PcMembers List, Paper* and *notification*.

In the organizational model, there are resources that depend on other resources to be created. In our case study, for example, a *Notification* is always linked with a *Paper,* and the existence of the *Notification* is conditioned by the action of generating a notification executed by the *PcChair*. These types of resources have a constant attribute (the name of the resource to which they are linked) in their specification. The rest of the resources of the organizational model do not include a constant attribute.

Table 3 shows the specification of the resources *Paper* and *Notification*.

Table 3. Specification of the resources *Paper* and *Notification*

Entity **Paper** Has title, Author, coauthors: SetOf[Author], abstract, status: String End Paper
Entity **Notification** Attribute Constant: Paper Has Paper, comments, evaluation: String End Notification

5.4 Specification of Links in Resource Dependencies

In the Tropos framework, the dependency links connect the resources with the *depender* and *dependee* actors. The links are represented using the concept of *Relationship* of the KAOS language. A *Relationship* is a subordinate object. The existence of its instances depends upon the existence of the corresponding object instances linked by the relationship [5]. The KAOS specification of a resource contains a) the resource name, b) the relation type (in this paper, we propose the relationships: *Send*, *Sentto, Sendby* or *Receive*) and c) the cardinalities of the links among the resource and the actors that send and receive it. In this way, the relationship is specified as Actor-Resource-Actor. One of the actors is obtained directly from the resource dependency with the system actor (Actor-Resource-System), as well as the resource. The other actor is obtained from the name of the actor included in the specification of the module. For example, in the resource dependency *Paper* of the case study shown in Figure 1, the *Author* sends the resource *Paper* to the Software System Actor. From the module *obtain paper*, we obtain the *PcChair* Actor as the original actor of the dependency. Therefore, the elements of this resource dependency are: *Author*, *Paper* and *PcChair*. In this example, the cardinalities of the links of the resource dependency are used to indicate that: a) an *Author* can send 1 or more *Papers*, b) a *Paper* could be sent by 1 or more *Author*s, c) a *Paper* is sent only to a *PcChair*, and finally, d) a *PcChair* can receive 1 or more *Papers*.

Table 4 presents the specification of the links in the *Paper* resource dependency between the *Author* and the *PcChair*.

Table 4. Specification of links in the *Paper* resource dependency

Relationship SubmittedBy Links Author {Role send, Card 1..*} Paper {Role sentby, Card 1..*}	Relationship SubmittedTo Links PcChair {Role receive, Card 1..*} Paper {Role sentto, Card 1..1}

5.5 Task Dependency Specification

The task dependencies are represented using the concept of *Action* in the KAOS language. The action specification in KAOS is composed of: a) input parameters that permit the execution of the actions b) The output parameters that represent the resources generated as a result of these actions and c) the pre and post conditions of the action.

In the proposal presented in this paper, the clause_*Concerns* was included in the Action definition of KAOS to link the task with the the *depender* and the *dependee* actors of the task dependency. The first actor of the clause indicates the *depender* (the actor who is dependent on another actor), and the second indicates the *dependee* (the actor on whom another actor depends) of the dependency relationship.

Table 5 presents the KAOS-like specification of the *Submit Paper* task dependency. In this example, it is possible to determine that the action *Submit Paper* generates the resource *Paper* as output of the action. For this reason, the input parameters of the actions are the values of the attributes of the paper, and the output is the paper created. The postcondition of the action is the assignation of the input parameters to the attributes of the paper.

Table 5. Specification of the *Submit Paper* task dependency

Action Submit Paper Input String {Arg title}, String {Arg topics}, SetOf[string] {Arg abstract}, String {Arg status}, SetOf[String]{Arg author}, SetOf[String]{Arg coauthors} Output Paper {Arg paper} Precondition RegisterAuthor (author) Postcondition paper.title = title and paper.topics = topics and paper.abstract = abstract and paper.status = "sending" and paper.author = authors and paper.coauthors = coauthors Concerns PcChair, Author End Submit Paper

5.6 Specification of Modules in the Software System Actor

As was commented in section 4, the modules represent the internal tasks which allow the actors to satisfy their dependencies. In our proposal, the modules are composed by internal tasks linked to more general tasks using the task decomposition links of the Tropos Framework. In our proposal, the internal tasks of the system actor arise as a result of moving *modules* from the organizational actors. For example, the internal task *sort papers* is one of the tasks that allow the software system to perform the task *send notifications to the Authors*.

The specification of internal tasks is similar to the specification of the task dependency, but the clause *RelatedGoal* is also added to the original specification of KAOS. The clause *RelatedGoal* allows us to create the task decomposition of the modules. Table 6 presents the KAOS-like specification of the task *sort papers,* which is related to the *send notifications to the Authors* module in the System Actor (see Figure 1).

Table 6. Specification of the *sort papers* task

Action SortPapers Input … Output .. Preconditions … Postconditions … Concerns System RelatedTask SendNotificationstotheAuthors End SortPapers

The application of these specification rules allows us to obtain the complete specification of the organizational model. This specification can be used to generate the specifications of the information system in an automatic way. To do this, it is necessary to define the correspondences between the elements of the organizational model and the elements of an object-oriented conceptual schema.

6 Conclusions and Further Work

Summarizing, in this paper, we present a methodological approach to generate the formal specification of the relevant information of a Tropos organizational model. To do this, we propose using the Tropos Framework [4], a well-known technique, to represent the organizational model. As a first step in the method, an organizational model which has been previously specified in the Tropos framework is selected. This Model is then enriched with the inclusion of the system actor. This facilitates the identification of the tasks to be automated using the information system. Finally, A method for translating each of the elements of the organizational model into formal specifications described in a KAOS-like language is also presented

We are now working on developing the translation rules to produce an object-oriented conceptual schema for the information System from the formal specification of the organizational model. By doing this, we go a step further in the process of including business modeling as a key piece in the software production process.

References

1. Andrade Luis, Amilcar Serdanas, Banking and Management Information System Automation, proceedings of the 13th World Congress of International Federation of Automatic Control, San Francisco, USA (1996), pp. 133-138.
2. Schwabe Daniel and Gustavo Rossi, An Object Oriented Approach to Web-Based Application Design, Proceedings of the Theory and Practice of Object Systems, New York, USA (1998), pp. 207-225.
3. Oscar Pastor, Jaime Gómez, E. Infrán, V. Pelechano, The OO-Method approach for information systems modeling: from object-oriented conceptual modeling to automated prgramming, Information Systems (IS), 26(7), (2001), pp. 507-534.
4. Castro J. Kolp M. Mylopoulos J. Towards Requirements-Driven Information Systems Engineering: The Tropos Project, Information System Journal, Elsevier, Vol 27 (2002), pp. 365-389.
5. Dardenne, A. Van Lamsweerde and S. Fickas, Goal Directed Requirements Acquisition, Science of Computer Programming, vol. 20, North Holland (1993) 3-50.
6. Yu, Eric, Modelling Strategic Relationships for Process Reengineering, Phd Thesis, University of Toronto, (1995).
7. Estrada H. Martínez A., +'Pastor O., Goal based business modeling oriented towards late requirements generation, Proceedings of the ER: Conceptual Modeling 04, Springer Verlag, Chicago, USA (2003), pp. 277-290.

A Weighted Fuzzy Min-Max Neural Network for Pattern Classification and Feature Extraction

Ho J. Kim[1], Tae W. Ryu[2], Thai T. Nguyen[2], Joon S. Lim[3], and Sudhir Gupta[4]

[1]School of CSEE, Handong University, Pohang, 791-940, Korea
hjkim@handong.edu
[2]Department of Computer Science, California State Univ., Fullerton, CA 92834, USA
tryu@fullerton.edu
[3] College of Software, Kyungwon University, Sungnam 461-701, Korea
jslim@mail.kyungwon.ac.kr
[4] Department of Medicine, University of California, Irvine, CA 92868, USA
sgupta@uci.edu

Abstract. In this paper a modified fuzzy min-max neural network model for pattern classification and feature extraction is described. We define a new hypercube membership function which has a weight factor to each of the feature within a hyperbox. The weight factor makes it possible to consider the degree of relevance of each feature to a class during the classification process. Based on the proposed model, a knowledge extraction method is presented. In this method, a list of relevant features for a given class is extracted from the trained network using the hyperbox membership functions and connection weights. For this purpose we define a Relevance Factor that represents a degree of relevance of a feature to the given class and a similarity measure between fuzzy membership functions of the hyperboxes. Experimental results for the proposed methods and discussions are presented for the evaluation of the effectiveness and feasibility of the proposed methods.

1 Introduction

The goal of pattern classification is to partition the feature space into decision regions. Artificial neural networks have been successfully used in many pattern classification problems [2], [4]. Fuzzy set theory was introduced by Zadeh [10] as a means of representing and processing data by allowing partial set membership rather than crisp set membership or non-membership. Fuzzy min-max (FMM) neural networks were introduced by Simpson [8] using the concept of hyperbox fuzzy sets. A hyperbox defines a region of the n-dimensional pattern space that has patterns with full class membership using its minimum point and its maximum point. The fuzzy min-max neural networks are built by making one pass through the input patterns and forming hyperboxes into fuzzy sets to represent pattern classes. Gabrys and Bargiela have proposed a General Fuzzy Min-Max (GFMM) neural network which is a generalization and extension of the FMM clustering and classification algorithm [3]. In GFMM method, input patterns can be fuzzy hyperboxes or crisp points in the pattern space.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 791–798, 2004.
© Springer-Verlag Berlin Heidelberg 2004

The learning process of FMM is simple but powerful. The FMM can add new pattern classes on the fly. It can refine existing pattern classes as new information is received, and it uses simple operations that allow for quick execution. However, regardless of how many training patterns contribute to each dimension of a final hyperbox in the neural network, the original FMM neural network does not give any weight to any of the dimensions. We present a modified FMM, called the weighted FMM (WFMM) neural network that takes the weights into account. The rationale for this idea is that a feature of a particular hyperbox can cover many more training patterns than other features of the same hyperbox and features of other hyperboxes. A weight value is assigned to each of the dimensions of each hyperbox so that membership can be assigned considering not only the occurrence of patterns but also the frequency of the occurrences within that dimension. The proposed WFMM can provide at least two advantages over the original FMM: (a) it would work better for pattern classification than the original scheme, especially for data sets with highly uneven distribution of features or noisy features since the hyperboxes in the WFMM model is not too sensitive to a few occurrence of unusual/noisy features in input patterns (b) the learned weights for each feature during training process can be used to identify the relevance of the feature to the given class, which can be easily used for possible rule generation [6], [9]. Particularly, the second advantage can be used for the important feature extraction in many applications. For example, suppose a doctor needs to diagnoses patients with various symptoms such as cough, fever, etc. It will be very useful for the doctor to know what symptoms (features) are the most relevant to a particular disease (a class) with possible range of expected values (e.g., temperature in the range of 37 ~ 39C). In this paper, we present a method of this type of feature extraction in the proposed WFMM model.

2 The FMM Neural Network

In the FMM model, a hyperbox defines a region of the n-dimensional hyper-space, and all patterns contained within the hyperbox have full cluster/class membership. Learning in the FMM neural network consists of creating, expanding and contracting hyperboxes in a pattern space. The membership function for an arbitrary hyperbox is defined as

$$\begin{aligned} b_j(A_h) = \frac{1}{2n}\sum_{i=1}^{n}[&\max(0,1-\max(0,\gamma\min(1,a_{hi}-v_{ji}))) \\ &+ \max(0,1-\max(0,\gamma\min(1,u_{ji}-a_{hi})))]. \end{aligned} \quad (1)$$

In the Equation (1), n means the number of features in the test pattern, γ is the sensitivity parameter in the range [0, 1] to control how fast membership value decreases as the distance between an input pattern and the hyperbox and increases. a_{hi} is the value of *i*-th feature of *h*-th input pattern. u_{ji} and v_{ji} mean the minimum and maximum value of dimension *i* of hyperbox b_j, respectively. The test pattern is said to

belong to the hyperbox if its membership value to the hyperbox is the highest compared to all other hyperboxes within the neural network.

3 The Proposed WFMM Model

In the proposed WFMM model, in addition to the hyperbox creation, expansion, contraction, the learning requires weight update. Unlike the FMM model, the membership function in the WFMM model has the weight factor to consider the relevance of each feature as different values. Equation (2) shows the hyperbox membership function for the proposed model. In the equation, w_{ij} is the connection weight between *i*-th feature and *j*-th hyperbox. The other notations are the same as Equation (1).

$$b_j(A_h) = \frac{1}{\sum_{i=1}^{n} w_{ij}} \bullet \sum_{i=1}^{n} w_{ij} [\max(0, 1 - \max(0, \gamma \min(1, a_{hi} - v_{ji}))) + \max(0, 1 - \max(0, \gamma \min(1, u_{ji} - a_{hi}))) - 1.0] \quad (2)$$

As compared to the original model, this membership function is modified in two ways. First, the weight factor W_{ji} is applied to each summation term. Second, the sum of complements of the min point violation and the max point violation is subtracted by 1.0 to make it fall in the range [0,1]. So, the (1/2n) factor used for the average is now replaced by the total weights to normalize the membership value. The original model logically considers each hyperbox dimension having a weight of 1.0. The result is that the density of any uneven pattern distribution is ignored. However, the WFMM model varies this weight based on training data. The weights are adjusted as Equation (3) and (4). The change to be made for the weight for i-th feature in j-th hyperbox is computed as Equation (4).

$$w_{ij}^{new} = w_{ij}^{old} + \Delta w_{ij} \quad (3)$$

$$\Delta w_{ij} = \begin{cases} \lambda & if\ (v^{new} - u^{new} \le s) \\ d \bullet (T - \frac{v^{new} - u^{new}}{v^{old} - u^{old}}) & else\ if\ (\frac{v^{new} - u^{new}}{v^{old} - u^{old}} \le T) \\ MAX(w_{ij}^{old} \bullet (\frac{v^{old} - u^{old}}{v^{new} - u^{new}} - 1.0), \frac{-w_{ij}^{old}}{2}) & Otherwise \end{cases} \quad (4)$$

Equation (3) and (4) show that the weight increases when more than two patterns that belong to a hyperbox, are given in the same area of the feature space. In the equations, parameter *s* is a standard size for feature range of hyperboxes. λ and *d* are learning rates which are small positive constants. $T \ge 1.0$ is used to control the learning process. In other words, λ and *d* are the factors to regulate how fast the weight of a dimension increases as it expands. The constant *T* is the threshold to determine weight increase or decrease. The initial value of each weight is 1.0 and the value is assigned when a new hyperbox is created.

In the proposed model, a method to reduce the effect of unusual/noisy patterns which may be included in the training data is considered. For the purpose, in the proposed

model we modified the expansion scheme so that the membership functions gradually increase for the unusual/noisy patterns. Equation (6) and (7) show the modified expansion scheme.

$$\text{If} \quad n\theta \geq \sum_{i=1}^{n} (\max(v_{ji,} x_{hi}) - \min(u_{ji}, x_{hi})) \tag{5}$$

Then If $(x_{hi} < u^{old})$

$$u_{ji}^{new} = u^{old} - \frac{1}{w_{ji}}(u^{old} - x_{hi}) \tag{6}$$

If $(x_{hi} > v^{old})$

$$v_{ji}^{new} = v^{old} + \frac{1}{w_{ji}}(x_{hi} - v^{old}) \tag{7}$$

4 Feature Extraction

The goal of feature extraction is to determine the relevance of each feature in a pattern class. For example, with the Iris plants data set [1], it is interesting to know which of the three features (sepal length, sepal width, and petal length) is the most relevant to each of the three classes (Iris Setosa, Iris Versicolour, and Iris Virginica). Furthermore, for each of the features, what typical range of feature values can be expected for a pattern class. The answer to the second question can be easily extracted from the training results by choosing the range values of the highest weight. For this purpose we define the *relevance factor* (RF) that represents the degree of relevance of a feature to a given class. We describe the feature extraction method in detail as follows.

Let each hyperbox fuzzy set, B_j, be defined by the ordered set $B_j = \{X, U_j, V_j, f(X, U_j, V_j)\} \quad {}^{\forall}X \in I^n$.

Using this definition of a hyperbox fuzzy set, the aggregated fuzzy set that defines the k-th pattern class C_k is defined as

$C_k = \bigcup_{j \in K} B_j$, where K is the index set of those hyperboxes associated with class k, $U_j = (u_{j1}, u_{j2}, \cdots, u_{jn})$ is the min point for B_j, and $V_j = (v_{j1}, v_{j2}, \cdots, v_{jn})$ is the max point for B_j.

From the trained network, the list of relevant features for a given class can be generated as follows. First as shown in Equation (8) we define a *Relevance Factor* (RF) of a feature representation f_i with respect to a class k.

$$RF(f_i, k) = \left(\frac{1}{N_k} \sum_{B_j \in c_k} S(f_i, (u_{ji}, v_{ji})) \cdot w_{ij} - \frac{1}{(N_B - N_k)} \sum_{B_j \notin c_k} S(f_i, (u_{ji}, v_{ji})) \cdot w_{ij}\right) / \sum_{B_j \in c_k} w_{ij} \tag{8}$$

Constant N_B and N_k are the total number of hyperboxes and the number of hyperboxes that belong to class *k*, respectively. In Equation (8), the feature f_i can be defined as a fuzzy interval value which consists of min and max values on the *i*-th dimension out of the *n*-dimension feature space. For an arbitrary feature f_i, let f_i^L and f_i^U be the min and max values, respectively, then the similarity measure S between two fuzzy intervals can be defined as Equation (9).

$$S(f_i,(u_i,v_i)) = S((f_i^L,f_i^U),(u_i,v_i)) = \frac{Overlap((f_i^L,f_i^U),(u_i,v_i))}{Max(f_i^U - f_i^L, v_i - u_i)} \qquad (9)$$

In Equation (9), if two fuzzy intervals are all point data, then the denominator part of the euqation, $Max(f_i^U - f_i^L, v_i - u_i)$ becomes zero. Therefore we define the similarity measure in this case as Equation (10). As shown in the equation, the similarity value is 1.0 when two interval are an identical point, and 0 when they indicate two different points.

$$S((f_i^L,f_i^U),(u_i,v_i)) = \begin{cases} 1 & if \quad (f_i^L = f_i^U = u_i = v_i) \\ 0 & Otherwise \end{cases} \qquad (10)$$

But if $Max(f_i^U - f_i^L, v_i - u_i)$ is greater than zero, the value is determined as described in Equation (11).

$$Overlap((f_i^L,f_i^U),(u_i,v_i)) = \begin{cases} f_i^U - u_i & if(f_i^L \le u_i \le f_i^U \le v_i) \\ v_i - u_i & if(f_i^L \le u_i \le v_i \le f_i^U) \\ f_i^U - f_i^L & if(u_i \le f_i^L \le f_i^U \le v_i) \\ v_i - f_i^L & if(u_i \le f_i^L \le v_i \le f_i^U) \\ 0 & Otherwise \end{cases} \qquad (11)$$

If the $RF(f_i,k)$ has a positive value, it means an excitatory relationship between the feature f_i and the class *k*. But a negative value of $RF(f_i,k)$ means an inhibitory relationship between them. A list of interesting features for a given class can be extracted using the value of RF computed for each feature.

5 Experimental Results

In order to compare the original FMM model and the proposed WFMM model, the following error measure is used.

$$E = \frac{1}{pm}\sum_{i=1}^{p}\sum_{k=1}^{m}\left|c_{ik} - d_{ik}\right| \tag{12}$$

In Equation (12), p is the number of test patterns, and m is the number of classes. The c_{ik} and d_{ik} are the actual output and desired output value of class k for the i-th input pattern, respectively. We performed several experiments on the two data sets, Fisher iris data set and Cleveland medical data set, which can be obtained from [1].

Experiment 1: Iris Data Classification

The data set consists of 150 pattern cases in three classes (50 for each class) in which each pattern consists of four features. The parameter tuning test has shown that one of the values for θ that yields the best classification as approximately 0.30. Although this value may depend on the specific splitting of patterns for testing and training, the same value of θ = 0.30 is selected for the classification test as shown in Table 1. In the table, the first column represents the number of training patterns out of 150 patterns.

We used the mixture of Random subsampling and leave-one-out methods for assessing classifier accuracy. Table 1 shows the classification results in terms of number of error patterns and the error rate defined in Equation (12). The results show that the proposed model is slightly better than the original FMM model in terms of the error rate. There is no performance difference in terms of misclassification rate. The effect of weights can only show up at high hyperbox maximum sizes, although misclassification often occurs at these high values. Consequently, in order for the proposed model to outperform the original scheme significantly, the data set should exhibit a highly uneven distribution of weights at a high value of hyperbox maximum size.

Table 1. Comparison of the classification performance for the original FMM model and the proposed WFMM model. Based on the parameter tuning result, we set θ = 0.3, γ = 0.5 for the original FMM model and T = 1.2, d = 0.1, s = 0.05 and l = 0.1 for the proposed WFMM model.

# of training patterns	**Original FMM**		**Proposed WFMM**	
	# of error patterns	error rate	# of error patterns	error rate
30	4	0.62277	4	0.57643
60	3	0.63080	3	0.59236
90	1	0.63094	1	0.59241
120	0	0.63222	0	0.59498
149	0	0.63720	0	0.60504

Experiment 2: Medical Diagnosis Data Classification

The data were collected by V.A. Medical Center, Long Beach and Cleveland Clinic Foundation: Robert Detrano, M.D., Ph.D. The data set consists of 297 pattern cases in five classes in which each pattern case has thirteen features. For the evaluation of classification performance with this data set, we set the same parameter values as used for Iris data set for the FMM model. On the other hand, a low value of T = 1.01 and high value of d = 30.0 are used to amplify the effect of the proposed WFM

model. Other parameter values are kept the same as with the Iris data. Table 2 shows the comparison of classification rates for original FMM with the proposed WFMM model.

Based on the parameter tuning result, we set $\theta = 0.93$, $\gamma = 0.5$ for the original FMM model and $T = 1.01$, $d = 30.0$, $s = 0.05$ and $l = 0.1$ for the proposed WFMM model. The first column represents the number of training patterns out of 297 patterns. Table 2 shows that the proposed WFMM model consistently performs better than the original FMM model in terms of mis-classification rate.

Table 2. Comparison of the classification performance for the original FMM model and the proposed WFMM model.

# of training patterns	**# of error patterns**	
	Original FMM	Proposed WFMM
10	7	5
20	14	12
30	21	19
40	29	27
50	35	32

The feature extraction algorithm described in the previous section has been tested on the same data sets used for classification test. Table 3 illustrates the list of features extracted for each class. In the table, feature names are represented as F1, F2, …, etc., as shown in the order of the original data set. For each data set, we picked only the first 4 features in the order of RF value because of the limited space

Table 3. A result of feature extraction for Cleveland medical data set

Pattern class	**List of relevant features with RF and Weight**
0	F12: (0.0,0.0), RF=0.00491, W=1.8, F9: (0.0,0.0), RF=0.00379, W=1.8, F3: (0.66667,0.66667) RF=0.00303, W=1.5, F2: (1.0, 1.0), RF=0.00317, W=1.8
1	F2: (1.0, 1.0), RF=0.00128, W=1.6, F1: (0.22917, 0.60417), RF=0.00122, W=2.81942, F12: (0.0, 0.0), RF=0.00119, W=1.3, F6: (0.0, 0.0), RF=0.001, W=1.6
2	F3: (1.0, 1.0), RF=0.00333, W=1.2, F6: (0.0, 0.0), RF=0.00285, W=1.2, F8: (0.24427, 0.54198), RF=0.00215, W=2.95092, F10: (0.22581, 0.45161), RF=0.00142, W=2.39988,

6 Conclusion

Fuzzy neural networks are hybrid system that have the advantages of both neural networks and fuzzy systems. While neural networks have learning ability and connectionist structure, the characteristics of fuzzy systems are human-like reasoning capacity and the ease of incorporating expert knowledge. In this paper, we proposed a

modified fuzzy neural network, WFMM based on the FMM taking the weight for each feature into account to improve the classification power for the data with unusual/noisy patterns and to extract interesting features for each class in a data set. In the proposed WFMM model, we defined the new hyperbox membership function and expansion scheme to consider the weight for each feature during learning process. By considering the weight for each feature, the proposed WFMM model becomes less sensitive to the unusual/noisy patterns in a data set than the original model. In addition, we defined a relevance factor to measure the degree of relevance for each feature to a given class. Using this information, the user can extract the list of features that are the most relevant to each class in a data set. This feature extraction method can be used for rule generation from a data set. The experimental results show that the proposed model, WFMM outperforms the original model, FMM for classification test—and effectively used in extracting interesting features for each class with the range of expected values.

Acknowledgement. This research was supported in part by Brain Science and Engineering Research Program sponsored by Korean Ministry of Science and Technology.

References

1. Blake, C.L. and Merz, C.J.: UCI Repository of machine learning databases [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, CA: University of California, Department of Information and Computer Science (1998)
2. Duda, R.O. and Hart, P.E.: Pattern Classification and Scene Analysis. New York: Wiley (1973)
3. Gabrys, B. and Bargiela, A.: General Fuzzy Min-Max Neural Network for Clustering and Classification. IEEE Transaction on Neural Networks, Vol.11, No.3, (2000)
4. Haykin, S.: Neural Networks, a comprehensive foundation. Prentice Hall, New Jersey (1999)
5. Mitra, S. and Hayashi, Y.: Neuro-Fuzzy Rule Generation: Survey in Soft Computing Framework. IEEE Transactions on Neural Networks, Vol.11, No.3, pp.748-768 (2000)
6. Mitra, S., De, R.K, and Pal, S.K.: Knowledge-Based Fuzzy MLP for Classification and Rule Generation. IEEE Transactions on Neural Networks, Vol.8, No.6, pp.1338-1350 (1997)
7. Nguyen, T.T.G.: A modified fuzzy min-max neural network for pattern classification and rule extraction. Department of Computer Science, California State University, Fullerton, Master Thesis (2003)
8. Simpson, P.: Fuzzy Min-Max Neural Networks – Part 1:Classification. IEEE Transaction on Neural Networks, Vol.3, No.5, pp.776-786 (1992)
9. Ye, C. Z., Yang, J., Geng, D., Zhou, Y., Chen, N. Y.: Fuzzy Rules to Predict Degree of Malignancy in Brain Glioma. Medical and Biological Engineering and Computing, Vol.40 (2002)
10. Zadeh, L.: Fuzzy sets. Information and Control, Vol. 8, pp. 338-353 (1965)

The eSAIDA Stream Authentication Scheme

Yongsu Park and Yookun Cho

Department of Computer Science and Engineering, Seoul National University,
San 56-1 Shilim-Dong Gwanak-Ku, Seoul 151-742, Korea
{yspark,cho}@ssrnet.snu.ac.kr

Abstract. To enable widespread commercial stream services, authentication is an important and challenging problem. There are three issues to consider for authenticating live streams: computation cost on the sender, communication overhead and verification probability on the receiver. As far as we know, SAIDA (Signature Amortization using IDA) is claimed to be the best algorithm in terms of the verification probability. In this paper, we describe eSAIDA, an efficient stream authentication scheme that is an improvement of SAIDA. We prove that under the restricted condition, the verification probability of eSAIDA is not less than that of SAIDA. Simulation results showed that under the same communication overhead its verification probability is much higher than that of SAIDA. Under various conditions, we measured the elapsed time of each scheme on the sender, which showed that the computation cost of the eSAIDA is lower than that of SAIDA.

Keywords: Network security, authentication, digital signature, stream distribution

1 Introduction

To enable widespread commercial stream services, it is crucial to ensure data integrity and source authentication [4,5,11], e.g., a listener may feel the need to be assured that news streams have not been altered and were made by the original broadcast station.

There are three issues to consider for authenticating live streams. For a sender, the scheme must have low computation cost to support fast packet rates. For receivers, it must assure a high verification probability, which is defined as a ratio of verifiable packets to received packets, in spite of a large packet loss [11]. Moreover, communication overhead should be small. In this regard, SAIDA (Signature Amortization using IDA) is claimed to be the best algorithm in terms of the verification probability [9].

In this paper, we propose an enhanced SAIDA (eSAIDA). We mathematically analyze that under the restricted condition, the verification probability of eSAIDA is not less than that of SAIDA. Simulation results showed that under the same communication overhead, the verification probability of eSAIDA is much higher than that of SAIDA. After we implemented SAIDA and eSAIDA algorithms, we measured the elapsed time on the sender under various conditions, which showed that eSAIDA is more efficient than SAIDA.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 799–807, 2004.
© Springer-Verlag Berlin Heidelberg 2004

This paper is organized as follows. Section 2 covers related work. In Section 3, we describe SAIDA and then propose our scheme, eSAIDA in detail. In Section 4, we analyze the verification probability on the receiver compared between SAIDA and eSAIDA. In Section 5, we shows the comparison results of the computation cost of the sender between the two schemes. Finally, conclusions are made in Section 6.

2 Related Work

In this section, we briefly review the previous work. Researches on stream authentication can be classified into two types. First, researches for designing faster signature schemes and second, researches for amortizing each signing operation by making use of a single signature to authenticate several packets [5].

2.1 Researches Related to the Fast Signature Algorithms

In [15], Wong and Lam proposed methods to speed up FFS (Feige-Fiat-Shamir) signature scheme [6] by using CRT (Chinese Remainder Theorem) [6], reducing the size of verification key, and using precomputation with large memory. They showed that the verification in the scheme was as fast as that of RSA with small exponent and the signing operation was much faster than other schemes (RSA, DSA, ElGamal, Rabin). Moreover, they extended FFS to allow "adjustable and incremental" verification, in which a verifier could verify the signature at different levels, so that he can verify it at a lower security level with a small computation cost, and later increase the security level with larger computation time.

In [4], Gennero and Rohatgi proposed the on-line stream authentication method by using one-time signatures. Compared with the ordinary signature schemes, one-time (or k-time) signature scheme shows much faster sign/verification rates. However, the size of one-time (or k-time) signature is proportional to the size of the input message and is quite large, i.e., the size of Lamport one-time signature [6] of the SHA-1 hashed message is about 1300 bytes. In [14], Rohatgi used k-time signature based on TCR (Target Collision Resistant) function, which reduced the size of the signature to less than 300 bytes.

Recently, Perrig proposed an efficient one-time signature scheme, BiBa, where the size of the signature was much smaller then those of the previous one-time signature schemes [10]. However, in this scheme the computation cost for signature generation is higher and the public key size is larger than those of the previous one-time signature schemes such as Merkle-Winternitz [7] or Bleichenbacher-Maurer [2].

2.2 Researches Related to Amortizing Each Signing Operation

To the best of our knowledge, the first stream authentication scheme was proposed by Gennaro and Rohatgi [4]. In this scheme, the kth packet includes the hash value of the $(k+1)$th packet and only the first packet is signed in each

packet group so each signing operation is amortized over several packets. However, it has two weak points: the first is that it does not tolerate packet loss because successive packets in the group cannot be verified when a packet loss occurs. Moreover, the sender must compute authentication information from the last packet in a group to the first packet. Therefore, it is unlikely this scheme will be used to authenticate on-line streams.

In [15], Wong and Lam proposed a method for amortizing signing operation by using Merkle's authentication tree [8]. In this scheme, the sender constructs an authentication tree for each group of stream chunks and signs the root's value. Each packet includes the signature of root's value and the values of siblings of each node in the packet's path to the root. A received packet can be verified by reconstructing the values of nodes in the packet's path to the root and comparing the reconstructed root's value with the value in the signature. This scheme has an advantage in that all received packets are verifiable. But, it requires large communication overhead because each packet involves a lot of hashes and a signature. Another disadvantage is that all the packets in a packet group cannot be computed and sent until the root's value is signed, which in turn incurs bursty traffic patterns. In [15], the authors proposed another scheme that uses authentication stars. The authentication star is a special authentication tree such that its $n + 1$ nodes consist of one root and n leaves. This scheme has the shortcomings described above, too.

Perrig, Canetti, Song, and Tygar proposed TESLA (Timed Efficient Stream Loss-tolerant Authentication) and EMSS in [11]. TESLA uses MAC (Message Authentication Code) to authenticate packets. A fixed time interval after sending the packet that contains MAC, the key used for generating the MAC is revealed. This method is very efficient in that it requires low computation overhead and communication overhead. But, it has limitations in that the clocks of the sender and receivers must be synchronized within the allowable margin. Moreover, it does not provide non-repudiation. Recently, Perrig proposed the broadcast authentication protocol using BiBa [10], which has similar constraints of TESLA, such as time-synchronization or not providing non-repudiation.

In EMSS, the hash value of kth (index k) packet is stored in several packets whose indexes are more than k. After sending all the packets for a packet group, the sender transmits the signature packet that has the hashes of last packets in the group and their signature. The algorithm is very simple but is non-deterministic in the sense that the indexes are chosen at random. Moreover, the verification of each packet can be delayed and the delay bound cannot be determined. Particularly, although a packet cannot be verified by using all received packets and signature packet in one group, it can be verified after receiving packets and signature packets in the next groups. To increase the verification probability, the authors take the size of group to be small. So, even though a packet is unverifiable through the use of the packets of one group, it can be verified by receiving the packets of the next groups until the delay deadline. Authors also suggested extended EMSS, which is similar to EMSS but it uses IDA (Information Dispersal Algorithm) [12] to increase the verification probability.

Golle and Modadugu pointed out the fact that packet loss occurs quite bursty on the Internet and proposed GM scheme [5]. They proved that the scheme can resist the longest single burst packet loss assuming that there is a bound on memory size of hash buffer and packet buffer for the sender. Because each packet has only two hashes on average, this scheme has low communication overhead. Moreover, they showed that it is close to optimally resist burst packet loss under the practical condition such as constraining the average capacity of the buffer on the sender or estimating the endurance on the longest average burst loss. It has deterministic algorithm and its computation cost on the sender is quite low.

Recently, Park, Chong and Siegel devised an efficient stream authentication scheme, SAIDA (Signature Amortization using IDA) [9]. As in extended EMSS, SAIDA is based on Rabin's IDA algorithm. They mathematically analyzed the verificaton probability of SAIDA. Moreover, the simulation result showed that under the same communication overhead, the verification probability of SAIDA is much higher than those of the previous schemes. Detailed description of SAIDA is in Section 3.1.

3 Improvement: An Enhanced SAIDA Scheme

In this section, we describe the original SAIDA and then propose our scheme, eSAIDA. We will use the following notations. $h(X)$ denotes a one-way hash function. $C||D$ is the concatenation of strings C and D. $|E|$ denotes the bit size of E. The sender and the receiver are denoted by S and R, respectively. $SIG_F(G)$ stands for the digital signature of G signed by a signer F.

We assume that a live stream is divided into fixed-size chunks M_1, M_2, ... For the first n chunks (we call this a *group*), S generates n packets which include authentication information. After sending all these packets, S repeats this procedure for the next group of n chunks. When R receives the packets for a group, R attempts to verify them. Because packets are processed in the unit of a group, we will explain how the packets are generated/verified for only a single group.

SAIDA is based on IDA (Information Dispersal Algorithm) [12], which consists of the following two modules. $Disperse(F, m, n)$ splits the data F with some amount of redundancy resulting in n pieces F_i $(1 \leq i \leq n)$, where $|F_i|$ is $|F|/m$. Reconstruction of F is possible with any combination of m pieces by calling $Merge(\{F_{i_j}|(1 \leq j \leq m),\ (1 \leq i_j \leq n)\}, m, n)$.

3.1 SAIDA

SAIDA [9] works as follows.

1. For stream chunks M_1, ..., M_n, S computes $H_i = h(M_i)$ $(1 \leq i \leq n)$.
2. S calculates $H_{1\sim n}$=$h(H_1||\cdots||H_n)$ and $SIG_S(H_{1\sim n})$.
3. S obtains F_1, ..., F_n by calling $Disperse(H_1||\cdots||H_n||SIG_S(H_{1\sim n}), m, n)$.
4. S generates the packets $P_i = (M_i, F_i)$ $(1 \leq i \leq n)$.

When at least m of these n packets $P_{i_j} = (M_{i_j}, F_{i_j})$ $(1 \leq j \leq m)$ are successfully transmitted to R, R can verify all the received packets as follows.

1. R first reconstructs $H_1||\cdots||H_n||SIG_S(H_{1\sim n})$ by calling $Merge(\{F_{i_j}|(1 \leq j \leq m)\}, m, n)$.
2. R computes $H_{1\sim n} = h(H_1||\cdots||H_n)$ and verifies $SIG_S(H_{1\sim n})$.
3. R verifies each chunk M_{i_j} by checking that $h(M_{i_j}) = H_{i_j}$.

3.2 eSAIDA

In eSAIDA, some of the packets contain a hash value and the average number of such packets is parameterized by s. Under the condition that n is even, eSAIDA works as follows (when n is odd, a pre-defined dummy value, M_{n+1}, can be used).

1. S computes $H_i = h(M_i)$ $(1 \leq i \leq n)$ and calculates $H_{2j-1\sim 2j}$=$h(H_{2j-1}||H_{2j})$ $(1 \leq j \leq n/2)$.
2. S computes $H_{1\sim n} = h(H_{1\sim 2}||\cdots||H_{n-1\sim n})$ and $SIG_S(H_{1\sim n})$.
3. By calling $Disperse(H_{1\sim 2}||\cdots||H_{n-1\sim n}||SIG_S(H_{1\sim n}), m, n)$, S obtains F_1, ..., F_n.
4. For generating each packet P_i $(1 \leq i \leq n)$, S first selects a random number k_i $(1 \leq k_i \leq n)$. If $k_i > s$, $P_i = (M_i, F_i)$. Or, if $k_i \leq s$, $P_i = (M_i, F_i, H_{i+1})$ in the case when i is odd (or $P_i = (M_i, F_i, H_{i-1})$ if i is even).

When at least m of these n packets P_{i_j} $(1 \leq j \leq m)$ are successfully transmitted to R, R is able to verify some of them as follows.

1. R reconstructs $H_{1\sim 2}||\cdots||H_{n-1\sim n}||SIG_S(H_{1\sim n})$ by calling $Merge(\{F_{i_j}|(1 \leq j \leq m)\}, m, n)$.
2. R computes $H_{1\sim n} = h(H_{1\sim 2}||\cdots||H_{n-1\sim n})$ and verifies $SIG_S(H_{1\sim n})$.
3. R tries to verify each chunk M_{i_j} $(1 \leq j \leq m)$ as follows. Consider the case when i_j is odd. If $P_{i_j} = (M_{i_j}, F_{i_j})$ and P_{i_j+1} is not received, R is unable to verify M_{i_j}. Or, if $P_{i_j} = (M_{i_j}, F_{i_j}, H_{i_j+1})$ or P_{i_j+1} is received, R can verify M_{i_j} by checking that $h(h(M_{i_j})||H_{i_j+1}) = H_{i_j\sim i_j+1}$, where H_{i_j+1} is obtained either from P_{i_j} or by computing $h(M_{i_j+1})$. In the case when i_j is even, if $P_{i_j} = (M_{i_j}, F_{i_j}, H_{i_j-1})$ or P_{i_j-1} is received, R can verify M_{i_j} by checking that $h(H_{i_j-1}||h(M_{i_j})) = H_{i_j-1\sim i_j}$.

4 Verification Probability

In this section, we compare the verification probability of eSAIDA and SAIDA. V_e and V_s denote the verification probability of eSAIDA and that of SAIDA, respectively. We prove that under the restricted condition, $V_e \geq V_s$.

C_e and C_s denote the overhead per packet in eSAIDA and SAIDA, respectively. Assume that $s = n$ in eSAIDA and that both algorithms have the same group size $(= n)$ and the same communication overhead $(C_e = C_s)$. Then, Theorem 1 holds.

Theorem 1. *Under the above assumptions, if* $C_e \geq 2|h()| + \frac{2|SIG()|}{n}$ *then* $V_e \geq V_s$.

Proof. Because $s = n$ in eSAIDA, all the packets contain a hash value. In this case R can verify all the received packets if the number of the received packets is not less than m. Let m_e and m_s denote the minimum number of the received packets for successful verification in eSAIDA and SAIDA, respectively. We will prove $V_e \geq V_s$ by showing that $m_e \leq m_s$. $C_s = C_e \geq 2|h()| + \frac{2|SIG()|}{n}$ implies that

$$\begin{aligned} 2|h()| + \frac{2|SIG()|}{n} - C_e &\leq 0 \\ n|h()| + |SIG()| - \frac{n}{2}C_e &\leq 0 \\ \frac{|h()|\big(n|h()| + |SIG()| - \frac{n}{2}C_e\big)}{C_e\big(C_e - |h()|\big)} &\leq 0 \\ \frac{(n/2)|h()| + |SIG()|}{C_e - |h()|} - \frac{n|h()| + |SIG()|}{C_e} &\leq 0. \end{aligned} \tag{1}$$

Note that $C_s = \frac{n|h()|+|SIG()|}{m_s}$. Because $s = n$, $C_e = \frac{(n/2)|h()|+|SIG()|}{m_e} + |h()|$. Hence, $m_s = \frac{n|h()|+|SIG()|}{C_s} = \frac{n|h()|+|SIG()|}{C_e}$ and $m_e = \frac{(n/2)|h()|+|SIG()|}{C_e-|h()|}$. By applying Inequality (1), $m_e \leq m_s$ and $V_e \geq V_s$. □

If $s < n$ in eSAIDA, overhead per packet and V_e become smaller than those in the case when $s = n$. However, if packet loss and packet receiving occur bursty, the difference in the verification probability would not be significant due to the following reason: If P_i is received, it would be likely that R receives P_{i+1} (or P_{i-1}) which in turn results in a successful verification of M_i.

Generally, packet loss in the Internet is quite bursty [3,16]. To compare eSAIDA with SAIDA under the burst packet loss, we conducted experiments. To simulate the general pattern of the packet transmission in the Internet, we adopt 2-state Markov model for generating packet loss [11,16]. According to [11, 9], we set the average length of consecutive packet loss to be 8. The simulation results are the average values from 10^5 independent simulation runs. Fig. 1 and 2 are the simulation results, which shows that with proper selection of s, V_e is much larger than V_s.

5 Computation Cost

In this section, we analyze the computation cost of SAIDA and eSAIDA on the sender. Let C_{hash}, $C_{Disperse}(|F|, m, n)$, and C_{sign} denote the computation cost of the $h()$, $Disperse(F, m, n)$, and $SIG()$, respectively. Table 1 shows the comparison results of the computation cost. Although eSAIDA requires more $h()$ operations than SAIDA, the input size of $Disperse()$ is smaller than that of SAIDA.

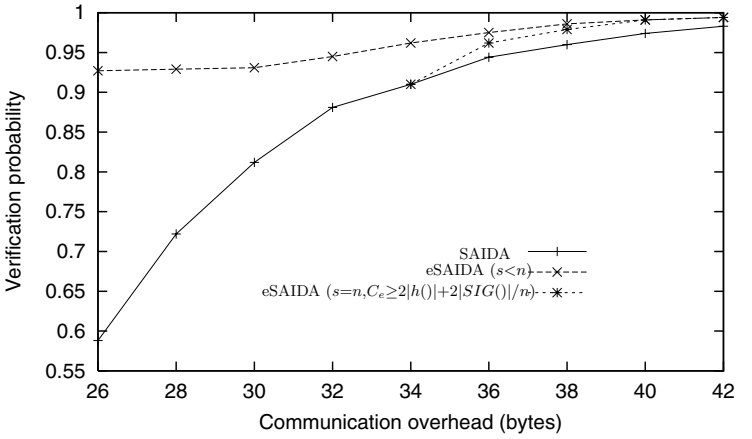


Fig. 1. Simulation results on the verification probability (n=128, packet loss rate: 0.35)

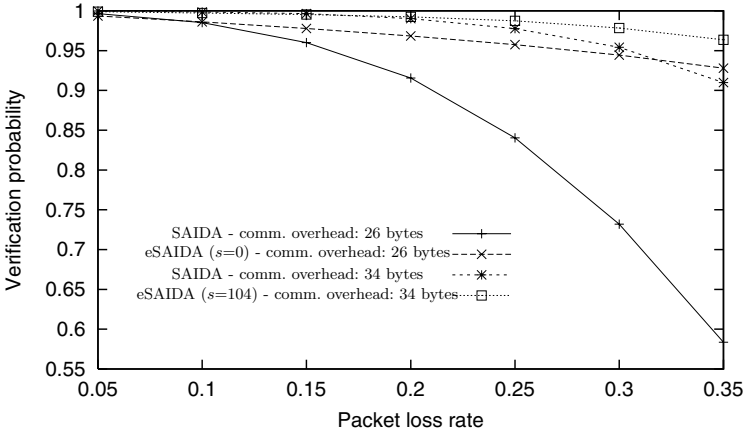


Fig. 2. Simulation results on the verification probability (n=128)

Table 1. Comparison results of the computation cost

Scheme	Computation cost for a single group
SAIDA	$(n+1)C_{hash}+$ $C_{Disperse}(n\|h()\|+\|SIG()\|,m,n)+C_{sign}$
eSAIDA	$(3n/2+1)C_{hash}+$ $C_{Disperse}(n\|h()\|/2+\|SIG()\|,m,n)+C_{sign}$

We implemented each scheme and measured the execution time under the conditions adopted from the following cases:

Table 2. Experimental results

Scheme	Elapsed time for a single group (ms)		
	[9]	Case I [11]	Case II [11]
SAIDA	78.9	140.5	873.5
eSAIDA	64.2	109.2	539.6

- **Case I [11]; Streamed Distribution of Traffic Data**: $|M_i| = 64,\ n = 200$.
- **Case II [11]; Real-time Video Broadcast**: $|M_i| = 512,\ n = 512$.
- **[9]:** $|M_i| = 512,\ n = 128$.

Detailed description on these cases is in [9,11]. Experimental environments are as follows: CPU, RAM, OS, crypto-library and compiler are Pentium 4 2.4 GHz, 512 MBytes, Linux 2.4.18, Crypto++ 4.2 and gcc 2.96, respectively. We use 160-bit SHA-1 [1,6] as the hash function and 1024-bit RSA [6,13] as the signature algorithm. The results are the average values from 10^5 independent simulation runs.

As can be seen in Table 2, the elapsed time of eSAIDA is 18%∼38% smaller than that of SAIDA. This is presumed to be due to the fact that the hash operation is much faster than $Disperse()$ that relies on finite field operations.

6 Conclusion

eSAIDA, which is an enhancement of SAIDA, showed higher verification probability and lower computation cost. We proved that under the restricted condition, the verification probability of eSAIDA is not less than that of SAIDA. Simulation results showed that under the same communication overhead, the verification probability of eSAIDA is much higher than that of SAIDA. Moreover, under the various conditions, the execution time of eSAIDA was 18%∼38% smaller than that of SAIDA.

References

1. FIPS 180-1. Secure Hash Standard. Federal Information Processing Standard (FIPS), Publication 180-1, National Institute of Standards and Technology, US Department of Commerce, Washington D.C., April 1995.
2. D. Bleichenbacher and U. Maurer. Optimal tree-based one-time digital signature schemes. In *STACS'96*, pages 363–374, 1996.
3. Michael S. Borella, Debbie Swider, S. Uludag, and G. Brewster. Internet Packet Loss: Measurement and Implications for End-to-End QoS. In *International Conference on Parallel Processing*, 1998.
4. Rosario Gennaro and Pankaj Rohatgi. How to Sign Digital Streams. In *CRYPTO'97*, pages 180–197, 1997.

5. Philippe Golle and Nagendra Modadugu. Authenticating Streamed Data in the Presence of Random Packet Loss. In *NDSS'01*, pages 13–22, 2001.
6. Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1997.
7. R. C. Merkle. A digital signature based on a conventional encryption function. In *CRYPTO'87*, pages 369–378, 1987.
8. Ralph C. Merkle. A Certified Digital Signature. In *CRYPTO'89*, 1989.
9. J. M. Park, E. K. P. Chong, and H. J. Siegel. Efficient Multicast Packet Authentication Using Signature Amoritization. *ACM Transactions on Information and System Security*, 6(2):258–285, 2003.
10. Adrian Perrig. The BiBa One-Time Signature and Broadcast Authentication Protocol. In *8th ACM Conference on Computer and Communication Security*, pages 28–37, November 2001.
11. Adrian Perrig, Ran Canetti, Dawn Song, and J. D. Tygar. Efficient Authentication and Signing of Multicast Streams over Lossy Channels. In *Proceedings of IEEE Security and Privacy Symposium*, May, 2000.
12. Michael O. Rabin. Efficient dispersal of information for security, load balancing and fault tolerance. *Journal of the Association for Computing Machinery*, 36(2):335–348, 1989.
13. R. L. Rivest, A. Shamir, and L. M. Adelman. A Method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.
14. Pankaj Rohatgi. A Compact and Fast Hybrid Signature Scheme for Multicast Packet Authentication. In *6th ACM Conference on Computer and Communication Security*, November 1999.
15. Chung Kei Wong and Simon S. Lam. Digital Signatures for Flows and Multicasts. *IEEE/ACM Transactions on Networking*, 7(4):502–513, 1999.
16. M. Yajnik, S. Moon, J. Kurose, and D. Towsley. Measurement and modelling of the temporal dependence in packet loss. In *IEEE INFOCOM'99*, 1999.

An Object-Oriented Metric to Measure the Degree of Dependency Due to Unused Interfaces

René Santaolaya Salgado, Olivia G. Fragoso Diaz, Manuel A. Valdés Marrero, Isaac M. Vásquez Mendez, and Sheila L. Delfín Lara

Centro Nacional de Investigación y Desarrollo Tecnológico
Interior Internado Palmira s/n Col. Palmira
Cuernavaca, Morelos, México
{rene,ofragoso,valdescompany,isaacvm,sldl79}@cenidet.edu.mx

Abstract. Object-Oriented frameworks are sets of classes designed to work together in order to offer generic solutions to many specific problems within the same application domain. A situation that often arises from the design of a framework is the interface dependency problem produced by interface inheritance when subclasses do not really need the interfaces. This problem negatively affects frameworks in their reuse and extension qualities. Although we know that this problem exists, we do not have a way to measure to what extent this problem affects frameworks. In this paper an object-oriented metric to measure the degree of dependency due to unused interfaces is proposed. Case studies are presented in order to show how this metric helps to detect when frameworks have a serious interface dependency problem. With this information a quantitative decision can be made to take care of the problem.

1 Introduction

The object-oriented paradigm provides many benefits such as reusability, decomposition of problems into easily understood objects, and also the assistance to perform future modifications or functionality extensions in already built systems. But the object-oriented software development cycle is not easier than the typical procedural approach. Therefore, it is necessary to provide reliable guidelines that one may follow to help ensure good object-oriented programming practices and write reliable code.

One such way of providing guidelines is the use of object-oriented metrics, which are considered a standard against which one can measure the effectiveness of object-oriented techniques in the design of a system. These metrics can be applied to analyze source code as an indicator of quality attributes [1].

One way of implementing a good object-oriented programming practice is the construction of frameworks. Frameworks are sets of classes working together in a certain problem domain, offering generic solutions which can be adapted to many specific problems within the same domain.

Because of their construction techniques, these frameworks have to be tested to ensure that the benefits from object-oriented programming are met, and of course, the right way to test a framework is using object-oriented metrics.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 808–817, 2004.
© Springer-Verlag Berlin Heidelberg 2004

In this paper an object-oriented metric that quantifies the degree of dependency due to unused interfaces is proposed. The degree of dependency is an important problem and must be dealt with because it directly affects the capability to make future modifications and functionality extensions of a framework.

This paper is organized as follows. In Section 2, basic definitions of our research and the mathematical background of object-oriented metrics are introduced. Section 3 provides a description of the interface dependency problem and the reduction of the framework reuse and extension properties due to this problem. Section 4 explains the metric proposed in this paper. Section 5 provides the practical examples of case studies using the proposed metric. The conclusions and directions for future work are discussed in Section 6.

2 Important Concepts

In this section the definitions of framework and software engineering metrics are presented. In addition, the metric properties applied to the proposed metric are explained.

2.1 Framework

Application frameworks are generally domain-specific applications, such as user interfaces, computer-integrated manufacturing frameworks, or multimedia collaborative work environments. A framework is more than a class hierarchy. It is a semi-complete application containing dynamic and static components that can be customized to produce user-specific applications. Due to the generic nature of framework components, mature frameworks can be reused as the basis for many other applications [2].

As we can see by the definition of framework, this software entity main objective is to allow reuse of the framework design and code, and this is achieved using inheritance and instantiation. Another objective of a framework is to facilitate the functionality extension of the framework, in order to meet certain requirements needed by a specific domain problem; this is also achieved using inheritance, adding concrete classes plugged into generic classes that offer the framework interfaces.

2.2 Software Engineering Metrics

These metrics are units of measurement that are used to characterize [1]:

- Software engineering products, e.g., designs, source code, and test cases.
- Software engineering processes, e.g., the activities of analyzing, designing, and coding.
- Software engineering people, e.g., the efficiency of an individual tester, or the productivity of an individual designer.

If used properly, software engineering metrics can allow us to [1]:

- Quantitatively define success and failure, and/or the degree of success or failure for a product, a process, or a person.
- Identify and quantify improvement, lack of improvement, or degradation in our products, processes, and people.
- Make meaningful and useful managerial and technical decisions.
- Identify trends.
- Make quantified and meaningful estimates.

2.3 Object-Oriented Metrics Characteristics and Properties

Object-Oriented metrics are different from typical procedural software metrics because of the localization, encapsulation, information hiding, inheritance, and object abstraction techniques [1].

Metrics for object-oriented programming are mostly classified into three classes: [3]

1. System-Level metrics: Such metrics consider number of files, trees, classes, children, class depth, and number of data variables, polymorphism, and multiple inheritance.
2. Tree metrics. Such metrics consider number of classes, tree depth, and number of member functions, number of data variables, polymorphism and multiple inheritance. The metric proposed in this paper is ranked in this category.
3. Class metrics. Such metrics consider number of subclasses, class depth, LOC (lines of code) of a class, number of member functions of a class, polymorphism, multiple inheritance, friend functions and operators, and operands.

In order to formally validate a metric several frameworks have been proposed, some of them are based on properties and others are based on the measurement theory. The first type is simpler than the second because the measurement theory is purely mathematical, a field not often understood by programmers and software developers.

To give some background to our metric, we choose one framework of each kind. The framework based on properties is the one proposed by Briand et al [4]. The framework based on the measurement theory is the one proposed by Zuse [3] and [5].

In the framework by Briand et al. [4] the metric is classified as cohesion metric. The demonstration of this assumption is beyond the scope of this paper. The concept of cohesion assesses the tightness with which related program features are grouped together into systems or modules. It is assumed that the better the programmer is able to encapsulate related program features together, the more reliable and maintainable the system will be [6]. The cohesion must be non-negative and, more important, to be normalized so that the measure is independent of the size of the system.

In the framework developed by Zuse [3] and [5] the metric covers the conditions of transitivity and completeness to be within the ordinal scale. Again, the demonstration is beyond the scope of this paper. The ordinal scale is unique only up to order. The admissible transformations are the monotonic increasing functions. Meaningful statistics are rank order statistics and non-parametric statistics. Units and the ordinal

scale are not compatible. The idea of units is to determine a difference or a quotient between numbers and the considered objects. Since the ordinal scale only considers ranking, units cannot be assigned.

As mentioned before, the metric is normalized. Normalization is a widely used method in order to get values between 0 and 1. Some consequences of this calibration of a metric are that statistics, as average, are not meaningful. But given our objective to measure the degree of dependency due to unused interfaces, these statistics are not needed, and we can assign an intuitive judgment to the numbers between 0 and 1.

3 Interface Dependency Problem

Some object oriented frameworks have the problem that their architecture makes their clients to depend on interfaces that are not used, and clients are affected when the interfaces are modified or new interfaces are added, which means that there is a high degree of coupling among the interfaces. This coupling should be avoided separating the interfaces wherever it is possible.

Interfaces are declared functions in base abstract classes without code whose children classes must implement, even with empty implementations because they do not need the interfaces.

The interface dependency problem appears in inheritance relationships where subclasses have a higher degree of coupling with its abstract base class. It is produced by the interface declaration in a class whose derived classes do not really need one.

This problem is shown in Fig. 1, where a small framework from a general domain is presented. The framework is about a double linked list, with elements of the floating type. The list is specialized to perform sorting and searching operations.

In the framework shown in Fig. 1, the sub-class named *Bubble* implements the bubble sort algorithm, and the sub-class named *Sequential* implements the sequential search algorithm. In order to access to the functionality of the framework, the client *Context* uses the interfaces from the *List* class *Sort( )* and *Search( )* inherited and implemented both by the sub-classes *Bubble* and *Sequential*.

Due to the framework structure, *Bubble* does an empty implementation of *Search( )* and *Sequential* does and empty implementation of *Sort( )*. Here is where the interface dependency problem appears due to unused interfaces.

Here we show that derived classes do not use all the inherited interfaces, and clients may get confused when they want to access to a given functionality since there is no indication as to which is the right interface to call, unless the client has a previous knowledge about the internal representation of the framework. This problem diminishes the framework reuse and extension capabilities, as we will explain below.

If a different client would not need the search functionality but just the sort functionality, it may not be possible to take only the sorting part to its application. This is because the class *Bubble* has an interface *Search( )* that has no reason to exist in the given context.

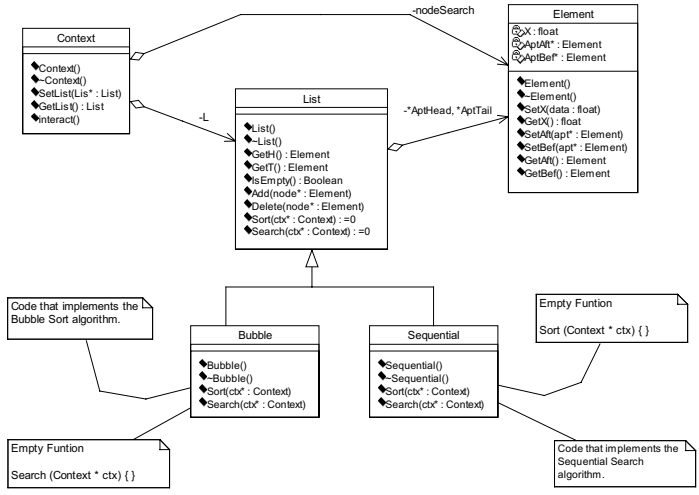


Fig. 1. Double linked list with two specialized classes; (*Bubble*), which implements the sorting of elements, and (*Sequential*), which performs the searching of elements. In the framework there are two interfaces (*Sort( )*) and (*Search( )*)

In addition, if a client needs extra functionality other than sorting and searching, say a printing algorithm, it will have to modify the *List* class to add the new interface *Print( )*, the *Bubble* and *Sequential* subclasses to add the same new interface, and the subclass which implements the new functionality will have to add the two existing interfaces *Sort( )* and *Search( )*. This is shown in Fig. 2.

The reuse property of the framework is diminished because we cannot take certain parts of the framework in an independent manner to another context. The extension property is diminished because we cannot extend the functionality of the framework without modifying the framework and, of course, this is not right or desirable in any object-oriented framework, whose ideal objective is precisely the reuse and extension facilities.

4 Proposed Metric V-DINO

In section 3 the interface dependency problem was shown, and the elements that intervene in the problem. In this section the elements that form the metric are first explained, and the metric is explained later.

The elements that intervene are the number of interfaces (NFV), the number of unused interfaces (NFNO), and the number of sub-classes (C-NOC).

- NFV. This number represents the interfaces declared within a given abstract class. This same number represents the number of interfaces that have to be implemented in each of the sub-classes derived from the abstract class.

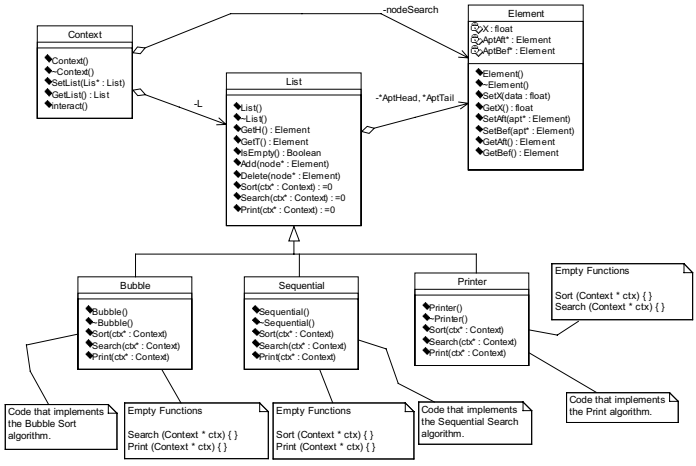


Fig. 2. New structure of the framework with extended functionality. A new class was added (*Printer*) with the new functionality. The interface (*Print( )*) with no code had to be added to the abstract class (*List*) and the two existing derived classes (*Bubble*) and (*Sequential*). Also the class (*Printer*) added the interfaces (*Sort( )*) and (*Search( )*) which are not used

- NFNO. This number represents the unused interfaces in all of the sub-classes of a given tree. For unused interface we understand an interface that has empty or null code only to meet the requirements of the programming language, but in a logical way has no reason to exist because it does not mean anything in the domain context of the framework.
- C-NOC. This number represents the number of immediate sub-classes subordinated to a class in the class hierarchy. For this purpose we will use the popular metric C-NOC (Chidamber - Number of Children) [7], which relates to the notion of scope of properties. It is a measure of how many sub-classes are going to inherit the methods of the parent class.

The proposed metric is called V-DINO (Valdes – Dependencia por Interfaces No Ocupadas – Unused Interface Dependency). This metric will measure the degree of the interface dependency problem due to unused interfaces, giving values in a normalized way. The mathematical expression is as shown in equation (1):

$$\text{V-DINO} = \text{NFNO} / (\text{C-NOC} \times \text{NFV}) . \quad (1)$$

As we can see, the expression C-NOC x NFV represents the total numbers of interfaces in all the sub-classes, whether they are needed or not.

Because of the above, the condition NFNO ≤ (C-NOC x NFV) will always be true for every framework, and therefore, $0 \leq \text{V-DINO} \leq 1$.

This metric can only be used for abstract classes that have at least one child, so that C-NOC ≠ 0 and NFV≠ 0. As mentioned in section 2.3, this metric is a tree metric, so that the analysis must be done separately for every tree in the framework. The tree must begin with an abstract base class.

The metric is in the ordinal scale, so its values can only be compared, not added or used in a mean. Because of the normalization, its values do not depend directly on the size of the framework or its trees.

The optimal value is 0, meaning that the problem does not exist in that tree, and 1 is the worst case (unreachable in practical cases) meaning that every interface is not needed. In the following section we show how the metric is used and how to interpret the value of V-DINO to take a good decision based on this value.

5 Case Studies Using V-DINO

In order to explain the usage of V-DINO we will use the frameworks from Fig. 1, Fig. 2, and Fig. 3.

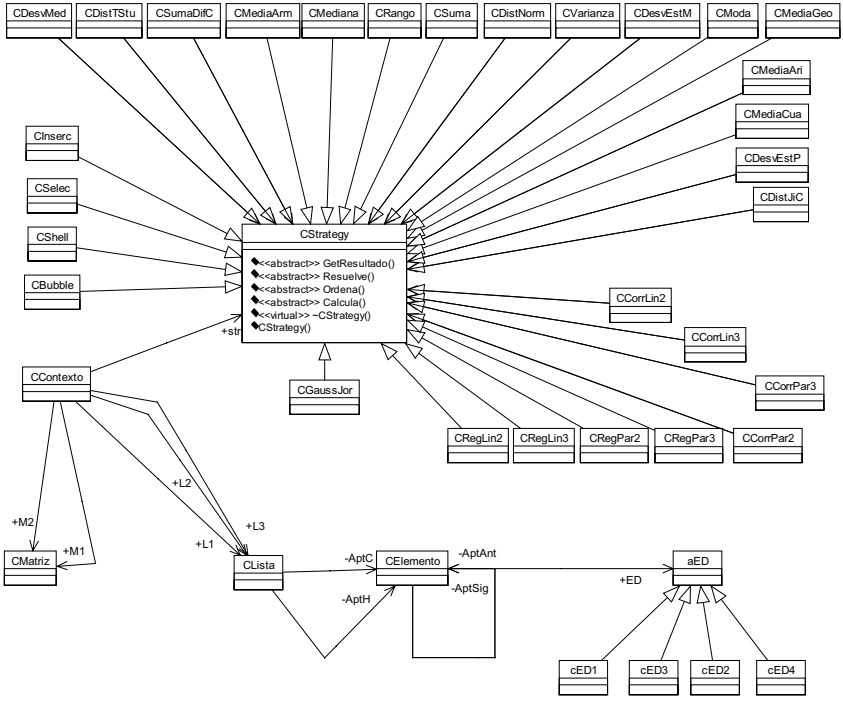


Fig. 3. Framework from the Statistics domain. There are two abstract classes (*CStrategy*) and (*aED*), which implement the interfaces (*GetResultado( ), Resuelve( ), Calcula( ), Ordena( )*) and (*GetX( ), SetX( )*) respectively

The framework shown in Fig. 3 is a real framework from the statistics domain, and presents two trees that can be analyzed using V-DINO. One tree has the statistics functionality and its abstract base class is called *CStrategy*, this tree does have the in-

terface dependency problem; the second tree has the list elements functionality and its abstract base class is called *aED*, this tree does not have the interface dependency problem.

By applying V-DINO to the frameworks from figures 1, 2 and 3, we obtained the results summarized in Table 1.

Table 1. Results for the elements involved in V-DINO calculus. (*NFV*) represents the number of interfaces, (*NFNO*) represents the number of unused interfaces in sub-classes, (*C-NOC*) represents the number of children and (*V-DINO*) represents the value of the metric itself

Framework	NFV	NFNO	C-NOC	V-DINO
Fig. 1 (List)	2	2	2	0.500
Fig. 2 (List)	3	6	3	0.666
Fig. 3 (CStrategy)	4	71	29	0.612
Fig. 3 (aED)	2	0	4	0.000

As shown in Table 1, all three frameworks have the interface dependency problem, when analyzed using the abstract class listed in the first column. As expected, the metric shows that in the tree of the *aED* class in figure 3, the problem does not exist.

With these V-DINO values, we can determine that all frameworks have serious problems, especially the one shown in Fig. 2. (0.666 is the highest value of the case study).

Using the metric, we experimentally have determined that the real problem begins with frameworks that have V-DINO ≥ 0.500 as we explain below.

The most common case of the interface dependency problem due to unused interfaces presents itself in frameworks like the ones shown in Fig. 1 and 2, where each derived class uses a different interface and the other ones are not needed. In these cases C-NOC = NFV and NFNO = (C-NOC x NFV) – NFV. For these cases, V-DINO is defined as shown in equation (2).

$$\text{V-DINO}=((\text{C-NOC} \times \text{NFV}) - \text{NFV})/(\text{C-NOC} \times \text{NFV}) = (\text{C-NOC} - 1)/\text{C-NOC} . \quad (2)$$

Therefore, according to equation (2), a framework with two classes like Fig. 1- has V-DINO= (2 – 1)/2 = 1/2 = 0.500, with three classes like Fig. 2- has V-DINO= (3 – 1) / 3 = 2/3 = 0.666, with four classes has V-DINO = (4 – 1) / 4 = 3/4 = 0.75, and so on.

Table 2 shows the behavior of V-DINO with a case, where each interface implementation is mutually exclusive in all sub-classes.

Table 2. Most common case of the interface dependency problem. The first column indicates the number of sub-classes (*C-NOC*) and the second column indicates the (*V-DINO*) metric value

C-NOC	V-DINO
2	1/2 = 0.500
3	2/3 = 0.666
4	3/4 = 0.750
5	4/5 = 0.800
6	5/6 = 0.833

As we can see in Table 2, V-DINO has a value above or equal to 0.500 and it will never reach the value of 1.

There are some algorithms [8, 9] to automatically solve the problem of interface dependency, however when V-DINO < 0.500 the problem cannot be solved using the algorithm presented in [8] because the interfaces are not mutually exclusive.

We say that the interfaces are mutually exclusive when; grouping together sub-classes that implement and use a specific interface, each sub-class is inside one and only one group. For example, if in Fig. 1 there was a third sub-class, different from *Bubble* and *Sequential*, that implemented and used both *Sort()* and *Search()*, then we would say that those interfaces are not mutually exclusive, because both are needed in the same class and, in this case, the interfaces do not need to be separated. This framework is shown in Fig. 4.

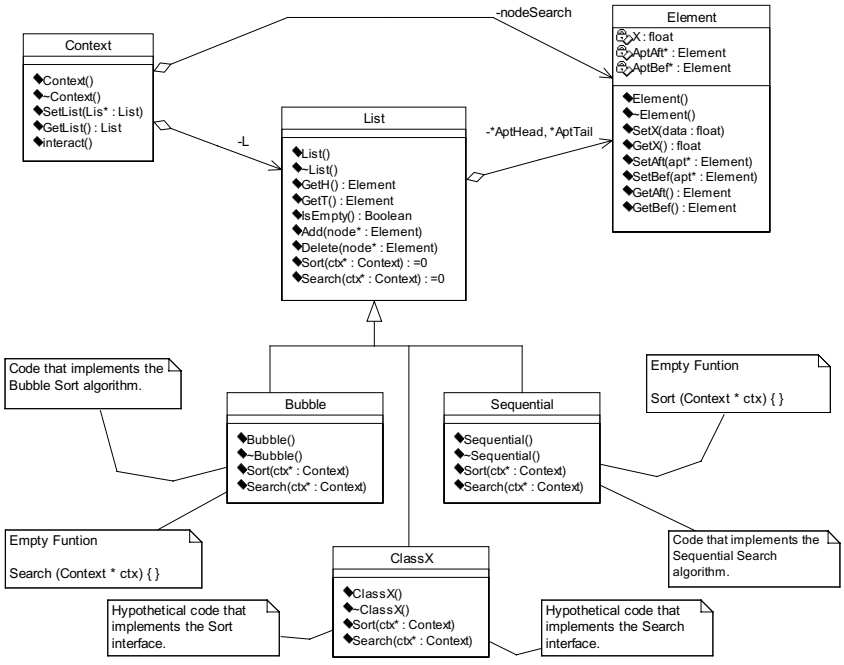


Fig. 4. Framework based on Fig. 1, adding a new sub-class (*ClassX*) that implements both (*Sort( )*) and (*Search( )*). This is an example of not mutually exclusive interfaces

For framework in the Fig. 4, we can clearly see that C-NOC=3, NFV=2 and NFNO=2, giving as a result that V-DINO=0.333. Because V-DINO < 0.500, we conclude that framework in Fig. 4 has a less severe interface dependency problem which does not need to be solved in the way proposed in [8]. Also, it is not considered a problem because both interfaces are needed; therefore those interfaces do not have to be separated.

6 Conclusions and Future Work

The metric presented in this paper can be used as quality software metric. In the ISO 9126 standard [10], which was established to characterize the quality of software, there are six attributes which software has to comply with in order to be considered quality software: functionality, reliability, usability, efficiency, maintainability and portability. The metric presented in this paper can objectively measure the changeability or extensibility degree which is a key factor in the maintainability of any software.

The ordinal scale is the base of software measurement [3], and the V-DINO metric fulfills the conditions of this scale. With those conditions met, we can compare trees within a framework to determine the degree of the interface dependency problem. With the added property of normalization we can easily make decisions to restructure the framework based on the value of the metric. With V-DINO we can also determine if the interface dependency problem exists within a certain framework.

The next step in this research is to create a tool capable of automatically calculate the metric, determine if the interface dependency problem exists and solve the problem in an automatic way using a refactoring tool. An algorithm for the solution of the problem is presented in [9] and a different algorithm is presented in [8].

We recommend the algorithm presented in [8] to be only used when V-DINO 0.500 and the other algorithm [9] to be used whenever V-DINO > 0, but this algorithm does not completely solve the problem and should be used only if V-DINO < 0.500. Note that [9] is a work in progress, and so this recommendation can change if the algorithm is modified by its authors.

References

1. Berand, E. V.: Metrics for Object-Oriented Software Engineering. The Object Agency, Inc. (2003)
2. Fayad, M. E., Johnson, R. E.: Domain-Specific Application Frameworks. Wiley Computer Publishing, John Wiley & Sons, Inc. (2000)
3. Zuse, H.: A Framework of Software Measurement. Walter de Gruyter, Berlin. (1998)
4. Briand, L. C., Morasca, S., Basili, V. R. 1996: Property-Based Software Engineering Measurement. In: IEEE Transactions on Software Engineering, no. 1, vol. 22. (1996) 68-85
5. Zuse, H.: Properties of Object-Oriented Software Measures. In: Proceedings of the Annual Oregon Workshop on Software Metrics (AOWSM), Silver State Park. (1995)
6. Fenton, N.: Software Measurement: A Necessary Scientific Basis. In: IEEE Transactions on Software Engineering, no. 3, vol. 20. (1994) 199-206
7. Chidamber, S. R., Kemerer, C. F.: A Metrics Suite for Object Oriented Design. In: IEEE Transactions on Software Engineering, no. 6, vol. 20. (1994) 476-493
8. Santaolaya, R., Fragoso, O. G., Valdés, M. A.: Refactorización de Frameworks por la Separación de Interfaces. In: Proceedings of the IEEE Reunión de Otoño de Comunicaciones, Computación y Electrónica (ROC&C'2003), Acapulco. México (2003)
9. Kerievsky, J.: Draft of Refactoring to Patterns v 0.17. Industrial Logic Inc. (2003)
10. ISO/IEC Standard 9126: ISO 9126 Software Product Evaluation - Quality Characteristics and Guidelines for their Use. Geneva (1991)

End-to-End QoS Management for VoIP Using DiffServ

Eun-Ju Ha[1] and Byeong-Soo Yun[2]

[1] Daegu Polytechnic College, 395 Manchondong-Dong, Susung-Gu, Taegu, Korea
ejha@mail.tpic.ac.kr
[2] Department of Electronic Engineering, Kyungpook National University, 1370 Sankyug-Dong, Buk-Gu, Taegu, Korea
bsyuntel@chollian.net

Abstract. The RTP/UDP/IP packet multiplexing scheme is combined with the DiffServ model to guarantee QoS deterioration problems over VoIP. The newly defined RTP/UDP/IP packets, namely, L_packets are proposed using DiffServ that are multiplexed at ingress routers to offer real-time communication services. To guarantee end-to-end QoS requirements, we obtain end-to-end delay requirements over a simple network topology. To prove the effectiveness of the proposed mechanism, we performed a simulation with varying the value of various parameters to evaluate the trend of the end-to-end QoS of the real-time voice traffics using network simulator (NS). The simulation results demonstrate that the proposed multiplexing scheme guarantees end-to-end QoS.

1 Introduction

Current voice over internet protocol (VoIP) packet transfer methods using low bit rate codes are still very inefficient due to the small payload size owing to the large overhead of the head size. These factors cause problems such as intolerable delay, jitter, and packet loss, all of which seriously deteriorate voice quality. There have been several related investigations dealing with the reduction of the overhead ratio using a real-time transport protocol/user datagram protocol/internet protocol (RTP/UDP/IP) packet multiplexing technique [1-3]. However, these previous attempts do not satisfy the true quality of service (QoS) requirements for real-time services.

In a real VoIP environment, which requires transmission of real-time traffic through packet network, end-to-end delay is considered to be the most important parameter to gauge QoS provision. The 150-ms one-way end-to-end delay limites defined by the international telecommunication union telecommunication standardization sector (ITU-T) G.114 recommendation [4] must be satisfied, including several different delay budgets such as propagation, serialization, and handling delay. Among these, queuing delay is only one component of the variable delay budget. Other components are fixed delay budgets [5].

In this paper, a new QoS guaranteeing mechanism is proposed that combines on RTP/UDP/IP packet multiplexing scheme with DiffServ QoS architecture.

Figure 1 shows the overall architecture of the proposed scheme including the packet multiplexing function. Newly defined RTP/UDP/IP multiplexing packets, namely, long packet (L_packet) that uses differentiated services (DiffServ) are multiplexed at ingress routers to offer real-time communication services. RTP/UDP/IP packet multiplexing scheme in [6] is adapted to our scheme. At

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 818–827, 2004.
© Springer-Verlag Berlin Heidelberg 2004

the ingress router, the appropriate control mechanism is required to support end-to-end QoS for input traffic. We solve this problem by using a DiffServ code point (DSCP).

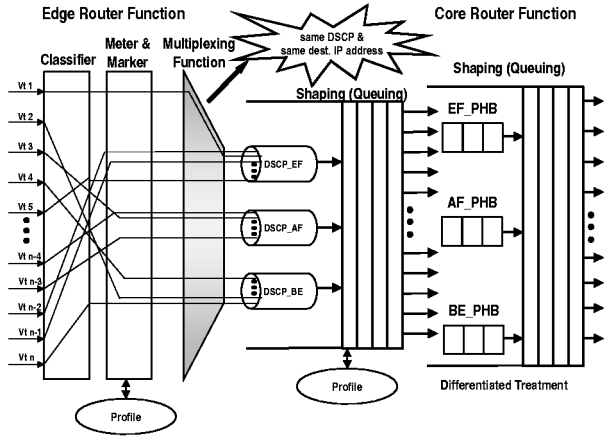


Fig. 1. Overall Architecture using L_packet

In an ingress router, each incoming packet is assigned to an appropriate DSCP before packet multiplexing. The packets with identical destination IP address and DSCP are multiplexed to the same L_packet. An ingress router contains a server, which is composed of DSCP_EF (DSCP Expedited Forwarding), DSCP_AF (DSCP Assured Forwarding), and DSCP_BE (DSCP Best Effort) queues. When excessive incoming voice traffics enter the ingress router simultaneously, packets are classified into different QoS guaranteed types by classifiers. Additionally, the qualified packets are queued in the DSCP_EF, DSCP_AF, and DSCP_BE server in turn. Using this L_packet format, we analyze the end-to-end delay of the proposed scheme. This scheme not only significantly reduces the short traffic flow management cost, but also guarantees the end-to-end QoS requirements. The remainder of this paper is organized as follows: Section 2, we present the L_packet format and detailed function of DiffServ Router. In Section 3, we implement the proposed multiplexing scheme using network simulator (NS). Our conclusion follows in Section 4.

2 VoIP System Architecture Using DiffServ

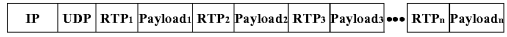
2.1 L_packet Format

In Figure 2, (a) shows the multiplexing packet format of [6]. It only consider the header compression and doesn't reflect differentiated QoS requirements of incoming traffics. In Figure 2, (b) shows the L_packet format, which adapt [6] to guarantee end-to-end QoS requirements. L_packet is used to the ingress, intermediate, and egress routers. At the ingress router, the L_packet, which has the same destination and DSCP, is classified and destined to the egress router through several intermediate routers. At the intermediate router, it only transits the L_packet without any

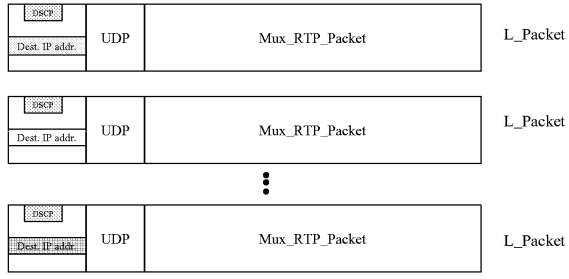
modification based on DSCP value. At the egress router, the arrived L_packets are separated and processed according to the priority of the DSCP value.

The key idea is as follows.

Add voice stream multiplexing scheme with identical destination IP address and DSCP (DiffServ Code Point)



(a) Previous RTP/UDP/IP Packet Format in [6]



(b) Proposed RTP/UDP/IP Packet Format using DiffServ

Fig. 2. Multiplexing RTP/UDP/IP Packet Format

2.2 Detailed Functions of DiffServ Router

Figure 3 shows traffic conditioning block (TCB) [7], which denotes the overall traffic processing procedure. When new voice traffics enter ingress router through an access network, it is classified twice. One is behavior aggregator (BA) classifier, and the other is multi-field (MF) classifier. The voice traffics through two classifiers enter into Meters. Meters are traffic conditioning elements that measure the rate of submitted traffic and compare it against a temporal profile. Meter might separate traffic simply into conforming and nonconforming traffic. The conforming traffics are marked a specific DSCP in a packet header and nonconforming traffics are re-marked. After that, the procedure of packet multiplexing is carried out.

Traffic Conditioner

The traffic conditioner measures the input traffic and assures that the packet behavior follows the predefined profiles. The traffic conditioner consists of a combination of meter, marker, multiplexer, shaper, and dropper. The DiffServ primarily conditions the traffic at the edge router only. Fig. 4 shows the traffic conditioner block.

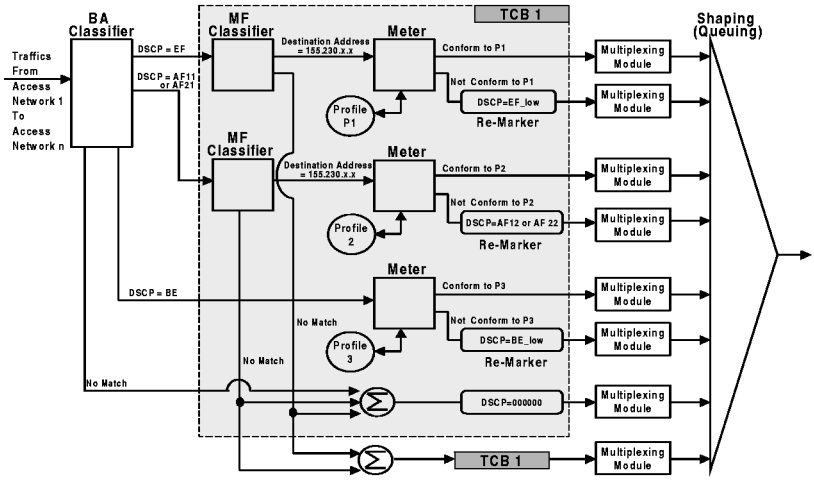


Fig. 3. Traffic Conditioning Block (TCB)

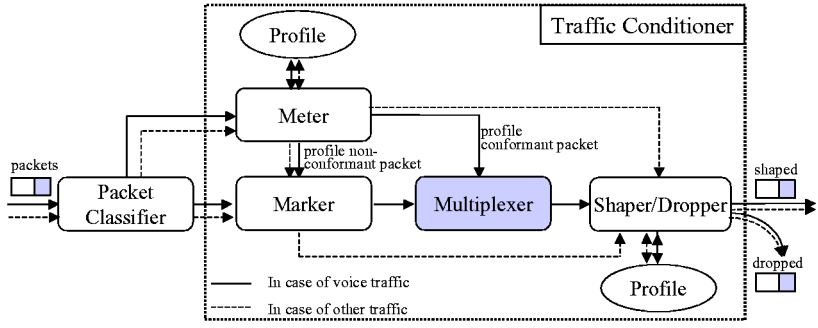


Fig. 4. DiffServ Traffic Conditioner Block

In Figure 4, we newly present multiplexer module. This module plays an important part of providing end-to-end QoS requirements. In case of other traffic, this module is not necessary. In case of voice traffic, this module is used to multiplex voice traffic with the same destination IP address as well as same DSCP value through classifier and meter module. Through this module, the large overhead size is greatly reduced as the number of multiplexing packet is increasing. Additionally, as the multiplexing criterion based on the same DSCP value, this also guarantees the QoS.

3 Experimental Results and Discussion

In this paper, we implement the proposed multiplexing scheme using network simulator (NS). The LNBL Network Simulator, *NS*, is a simulation tool developed by the Network Research Group at the Lawrence Berkeley National Laboratory. *NS* is an

extensible, easily configured and programmed event-driven simulation engine, with support for several flavors of TCP (include SACK, Tahoe and Reno) and router scheduling algorithms. *NS* is the simulator of an object-oriented concept, written in C++, with an Otcl interpreter as a front-end.

3.1 Implementation

We implemented our multiplexing scheme in *DiffServ* network using the *NS* DiffServ patch [8].

3.1.1 Overview of Packet Multiplexing Scheme

We propose the new RTP/UDP/IP packet multiplexing scheme for improving the VoIP performance using DiffServ and implemented the new *NS* components for simulating the proposed scheme with modifying the *NS* DiffServ patch software [8]. The basic behavior of the packet multiplexer is implemented in C++ language and the control and trace of those is described in tcl language. In Fig. 5, we present the flow chart of our packet multiplexing scheme and use this chart for implementing the *NS* simulator.

3.1.2 Developing DiffServ Components with Packet Multipexer

The software extends the elements of the *NS* network simulator to enable DiffServ networks to be simulated with multiplexing the incoming packet for real time traffic.

There are three key components to the extensions in *NS* DiffServ.

(1) The IP packet header has been modified to include a DiffServ codepoint (DSCP). The DSCP can have a value that is specified within the IETF; so, DSCPs such as EF, AFx1, Afx2, BE, etc. are used.
(2) DiffServ multiplexer (dsmux) and demultiplexer (dsdemux) components have been added. The dsmux is the similar component with the conditioner component except that the incoming packets are multiplexed up to the specified number. Each time a packet passes through a dsmux, the set of profiles within the dsmux is scanned and if the packet matches one of the profiles, then it is checked for conformance with the profile. If it is non-conformant, then some action is taken; EF packet is dropped and AFx1 traffic is remarked down to AFx2. Otherwise it is passed on unmodified. EF packets are sent to the multiplexing procedure to collect the packets with the same destination address up to the specified multiplexing number and sent to the next target node. Additionally, we implement the handling module of various codec for the change of the payload data size.
(3) A scheduler has been added. This scheduler consists of three different queues: each of the EF, AF, and BE traffic classes. These queues are serviced using a simple weighted round robin (WRR) scheduling algorithm.

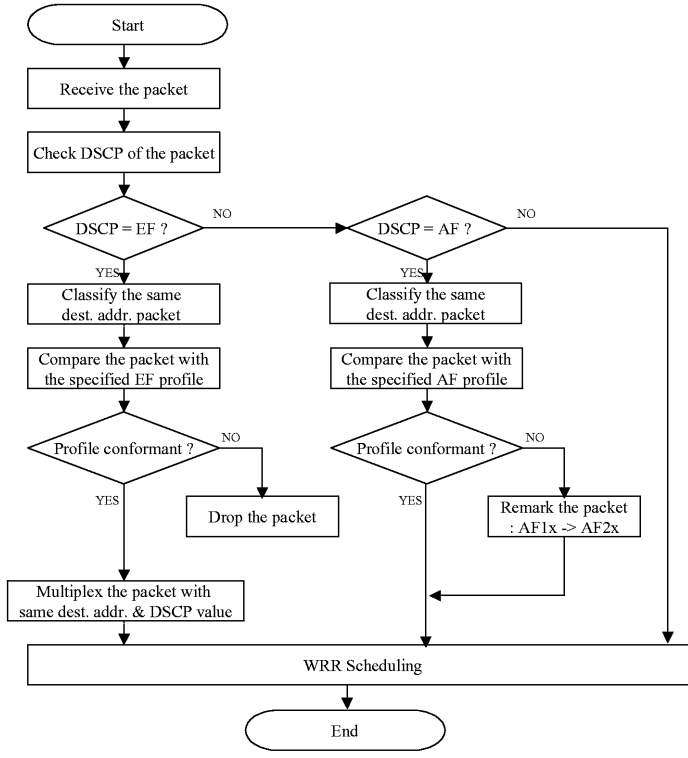


Fig. 5. Flow chart of packet multiplexing

These are the three main components that have been added. We implement the tool for calculating the queueing delay of the specified packet flow ID from the trace file. The queueing delay happens between the ingress router and the egress router. We assign the flow ID to each packet and monitor each packet's incoming time and outgoing time in the queue. The difference between the incoming time and the outgoing time of packet means the packet's duration time in the queue or the queueing delay. There are additional tcl scripts for control of the *NS* and DiffServ components above.

3.2 Simulation Results and Discussions

3.2.1 Simulation Topology and Its Element

In this section, we present the experimental results using NS. It is simulated based on [8] and modified for adapting the proposed L_packet format. The simulation model used is shown in Fig. 6. For simulation, we use a simple dumbbell topology [8]. For example of this topology, there may be the network connection between a telephone/computer user's site and central office of banking with the bottleneck on the WAN link. It may also be viewed in a more general sense, since many have argued; there is always a single bottleneck link on any network path. Such bottleneck

links are usually not fast moving. This justifies our choice of the network topology as a starting point to understand the impact (advancement) of real-time service of the proposed multiplexing scheme.

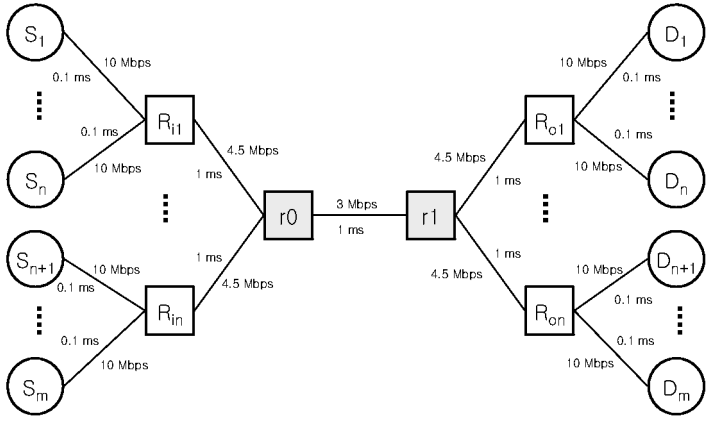


Fig. 6. Simulation Model

In our simulation, there are 2*m (m>0)* hosts on either side of a 3 Mbps bottleneck link with 1 msec delay; *m* of the hosts are sources and the other *m* are destinations with access link of 10 Mbps bandwidth and 0.1 msec delay. The *m* source hosts are composed of EF, AF, and BE traffic source generators. Queueing delay mainly occurs between r0 and r1. The 3 Mbps bottleneck link has the scheduler to contain 3 queues; one for EF, one for AF, and one for BE traffic. There are 3 different queue implementation, the EF queue to be a simple drop-tail queue, the AF queue to be an random early drop routers with in/out bit (RIO) queue, and the BE queue to be random early drop (RED) queue. We modified the enque method of a simple drop-tail queue for supporting RTP/UDP/IP packet multiplexing scheme. The scheduler of the bottleneck link serves these queues using WRR scheduling algorithm.

We consider homogeneous flows for real-time voice traffic where voice traffic aggregates consist of packets generated by sources which use the same codec algorithm [8]. In all the experiments, if EF is used to transport the voice traffic aggregates, then the EF traffic flows are generated to fill the subscribed rate of the EF class. Since the share of the bandwidth on the bottleneck link for EF class is fixed, the number of flows in the voice aggregate will change depending on the codec used. One reason for considering a fixed share of the bandwidth for voice traffic aggregates is that link bandwidth provisioning in the core of the network is usually done in bandwidth chunks (voice trunks).

3.2.2 Simulation Results

We have simulated the network model in Fig. 6 with varying the value of various parameters to review the trend of the end-to-end QoS of the real-time voice traffics. These parameters include the number of L_packet multiplexing count, the capacity of bottleneck link, the peak rate and the token bucket size of the profile, the weighting factor of the WRR scheduler and the queue length of the WRR scheduler. The

parameters used in our simulation are described in Table 1. The simulation time wass set to 100 sec, which consists of 20 sec warm up time for packet-stabilization.

Table 1. Defined Parameter Description

Parameter	Description	Default value
R	Peak rate of EF profile	400 kbps
B	Bucket size of EF profile	10000
W	Weighting factor of WRR	1:4:5
M	Number of multiplexing packet	20
T	Bottleneck (Trunk) link bandwidth	3 Mbps
C	Codec rate	6.4 kbps

Figure 7 and Figure 8 depict the queueing delay of each packet, which happens in the queue of bottleneck, with the time elapse. From these figures, the queueing delay of each packet has a value of less than 10 msec, which is the boundary of the real-time voice traffic delay. The increase the number of packet multiplexing causes the decrease of the number of marked points in the graph, which means the increase of the payload size with the fixed header size of each packet and the increase of the interarrival time at the ingress router. Additionally, greater the number of packets being multiplexed, the larger the maximum and minimum delay of the packet and there is an almost fixed packet delay variation.

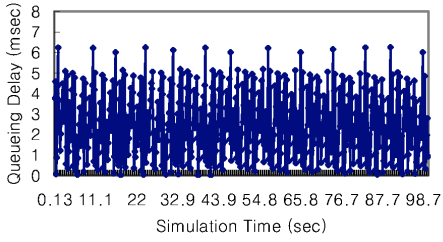


Fig. 7. Queueing Delay(Packet Multiplexing No.=8)

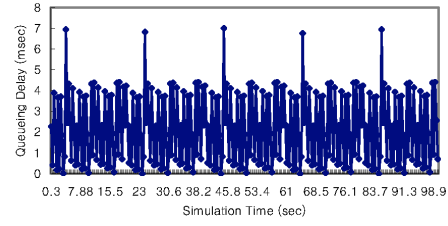


Fig. 8. Queueing Delay(Packet Multiplexing No.=16)

Figure 9depicts the average queueing delay of each incoming VoIP packet at the increase of the number of packet multiplexing. From this figure, we can see that the delay value of VoIP packet increases packet multiplexing increases. The increase packet multiplexing causes the larger length of the VoIP packet and the waiting time of the multiplexed VoIP packet rises in the queue of fixed length using the WRR scheduling as presented in Figure 9.

The threshold of the RTP/UDP/IP multiplexing packet is dependent upon the bucket size of the simulation. As time passed, the average queuing delay increased sluggishly. If the number of RTP/UDP/IP packet multiplexing is 40, the overhead ratio compared with no multiplexing packets reduces to 45.6 %. Thus, as the number of packet multiplexing increases, the overhead ratio is greatly reduced with a small increase in the average queuing delay. As previously noted, the queueing delay of the real-time voice traffic is between 6 msec and 10 msec.

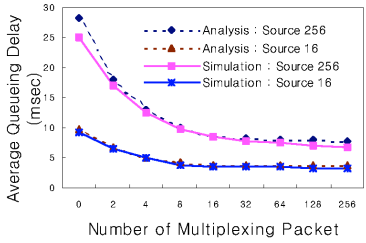


Fig. 9. Delay as a Function of the Mux Count

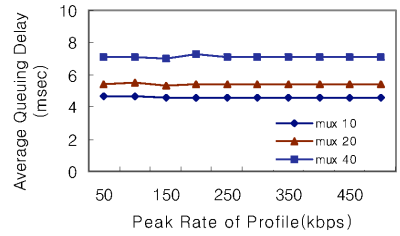


Fig. 10. Average Queueing Delay as a Function of Peak Rate of Profile

Fig. 9. shows that the appropriate number of packets multiplexed is less than 70, thus satisfying this boundary. Figure 10 depicts the average queueing delay of the multiplexed real-time packet at increasing the peak rate of EF profile. Fig. 10 shows that the average queueing delay of the multiplexed packets slightly fluctuates and has a random change pattern with the increase of the peak rate of EF profile. The peak rate of EF profile has little influence on the average queueing delay.

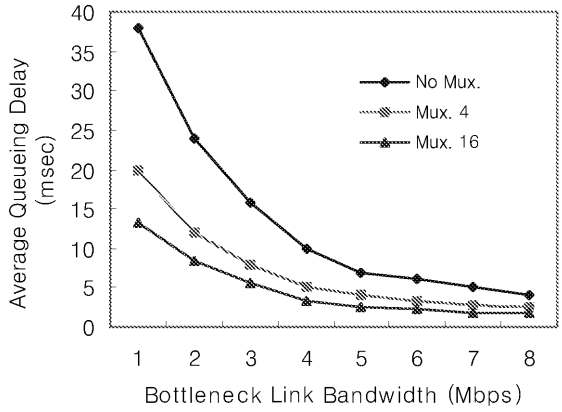


Fig. 11. Delay as a Function of the Bottleneck Link Bandwidth

Figure 11 depicts the queueing delay of real-time voice packet at increasing the bottleneck link bandwidth. As shown in this figure, the average queueing delay decreases logarithmically as the bandwidth of bottleneck link increases from 1 Mbps to 9 Mbps with the step of 1Mbps. The increase of bottleneck link means that more path is proved to service VoIP traffic. In Figure 11, we can find the asymptotic line of delay of about 2 msec when the bottleneck link has a very large value. Thus, there may be the trade-off between the costs of the bottleneck link which may be the leased line provided from the ISP and the performance of the average queueing delay. We also find again that an increase of the number of multiplexing packets cause an increase of the average queueing delay of packets.

4 Conclusion

We present the end-to-end QoS guaranteed voice traffic multiplexing scheme between VoIP access routers using DiffServ. At ingress router, the newly defined RTP/UDP/IP packets, namely, L_packet are multiplexed according to same destination egress router and same DSCP. We have implemented the proposed multiplexing scheme using NS. We have simulated with varying the value of various parameters to review the trend of the end-to-end QoS of the real-time voice traffics. These parameters may be such as the number of L_packet multiplexing count, the capacity of bottleneck link, the peak rate and the token bucket size of the profile, the weighting factor of the WRR scheduler, and the queue length of the WRR scheduler. The simulation results proved that the proposed multiplexing scheme could guarantee the end-to-end QoS.

Reference

[1] Katsuyoshi IIDA, Tetsuya TAKINE, Hideki SUNAHARA, and Yuji OIE: "Delay Analysis for CBR traffic in static-priority scheduling: single-node and homogeneous CBR traffic case," IEEE SPIE'97.

[2] D.De Vleeschauwer, J. Janssen, G. H. Petit, "Delay Bounds for Low Bit Rate Voice Transport over IP Network," Proceesings of the SPIE Conference on Performance and Control of Network Systems III, vol. 3841, pp. 40-48, September, 1999.

[3] Hassan Naser, and Alberto Leon-Garcia, "Voice over Differentiated Services," IEEE draft, draft-naser-voice-diffserv-eval-00.txt, December 1998.

[4] ITU-T Recommendation G.114, "One-Way Transmission Time."

[5] Bill Douskalis, *IP Telephony: The Integration of Robust VoIP Services*, Prentice Hall PTR

[6] Tohru Hoshi, Keiko Taniwaqa, and Koji Tsukada: 'Voice Stream Multiplexing between IP Telephony Gateways', IEICE Trans. INF. & SYST., Vol, E82-D, No. 4, April 1999.

[7] Yaram Bernet, *Networking Quality of Service and Windows Operating Systems*, New Riders.

[8] Sean Murphy, http://www.teltec.dcu.ie/~murphys/ns-work

Multi-modal Biometrics System Using Face and Signature

Dae Jong Lee, Keun Chang Kwak, Jun Oh Min, and Myung Geun Chun

Dept. of Electrical and Computer Engineering, Chungbuk National University, Cheongju, Korea
mgchun@chungbuk.ac.kr

Abstract. In this paper, we propose a multi-modal biometrics system based on the face and signature recognition. For this, we suggest biometric algorithms for the face and signature recognition. First, we describe a fuzzy linear discriminant analysis (LDA) method for the face recognition. It is an expanded version of the Fisherface method using the fuzzy logic which assigns fuzzy membership to the LDA feature values. On the other hand, the signature recognition has the problem that its performance is often deteriorated by signature variation from various factors. Therefore, we propose a robust online signature recognition method using LDA and so-called Partition Peak Points (PPP) matching technique. Finally, we propose a fusion method for multi-modal biometrics based on the support vector machine. From the various experiments, we find that the proposed method renders higher recognition rates comparing with the single biometric cases under various situations.

1 Introduction

With the development of information technology, the field of security is becoming more and more concerned. Under the information society, unauthorized user often destructs the information systems and then unveils the privacy and spread the unsound information. To tackle these problems, the biometrics is emerging as a promising technique. In the biometrics, we usually have studied iris, facial image, fingerprint, signature, and voiceprint. Among them, the face recognition is the most natural and straightforward method to identity each person.

This face recognition has been studied in various areas such as computer vision, image processing, and pattern recognition. The popular approaches for face recognition are PCA (Principle Component Analysis) [1] and LDA (Linear Discriminant Analysis) [2] methods. However, the major problem with the use of above methods is that they can be affected by variations of illumination condition and facial expression. Therefore we adopt a fuzzy LDA method for face recognition to improve the performance. One the other hand, the signature has been a familiar means where it is used for a personal authentication such as making a contact. The studies for signature recognition can be divided into online and offline ones. Here, the online signature recognition methods roughly belong to one of a global feature comparison, a point-to-point comparison and segment-to-segment comparison method [3]. For a

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 828–837, 2004.
© Springer-Verlag Berlin Heidelberg 2004

signature, however, contrary to other feature of biometrics, it has the problems that its skilled forgery is more or less easy and system performance is often deteriorated by signature variation from various factors [4]. Therefore, we propose a robust online signature recognition method using LDA and so-called Partition Peak Points(PPP) matching techniques. And finally, we propose a multimodal biometric system and suggest a fusion method based on the SVM(Support Vector Machine) which is capable of non-linear classification.

This paper is organized as follows. Section 2 describes the face recognition system using the new fuzzy-based LDA. In Section 3, we describe the signature recognition system using the LDA and PPP matching techniques. In Section 4, we explain the multimodal biometric system and decision rule based on SVM. Section 5 presents experiment results obtained for Chungbuk National University (CNU) face database and signature database, respectively. Finally, some concluding remarks are given in Section 6.

2 Face Recognition Using Fuzzy-Based LDA Method

The Linear Discriminant Analysis (LDA) is used to find optimal projection from feature vectors of face. Rather than finding a projection that maximizes the projected variance, LDA determines a projection, $V = W_{FLD}^{T} X$ ($\mathrm{W}_{\mathrm{FLD}}^{\mathrm{T}}$ is the optimal projection matrix), that maximizes the ratio between the between-class scatter and the within-class scatter matrix. However, this method of face recognition uses crisp class information for the given face images. One the other hand, the fuzzy-based LDA method assigns feature vectors to fuzzy membership degree based on the quality of training data. The procedures to assign a fuzzy membership degree to the feature vector transformed by PCA is as follows.

[Step 1] Obtain the Euclidean distance matrix between feature vectors of training sets.

[Step 2] Set diagonal elements to infinite (large value) in the distance matrix because of zero value in i=j case.

[Step 3] Sort the distance matrix in ascending order. And then, select the class corresponding to from i to k'th nearest point.

[Step 4] compute the membership grades for j'th sample point using the following equation.

$$\mu_{ij}(x) = \begin{cases} 0.51 + 0.49(n_{ij}/k), & if \quad i = the\ same\ as\ the\ label \\ 0.49(n_{ij}/k), & if \quad i \neq the\ same\ as\ the\ label \end{cases} \tag{1}$$

The value n_{ij} is the number of the neighbors belonging to the i'th class in j'th data. And then, we can calculate new feature vectors by using LDA based on fuzzy membership as shown in equation (1). The optimal k value in computing FKNN(Fuzzy K-Nearest Neighbor) initialization is determined by value representing the best recognition rate through each experiment.

The mean value of each class $\tilde{\mathbf{m}}_i$ is calculated by using feature vectors transformed by PCA and the fuzzy membership degree expressed in equation (1) as follows.

$$\tilde{\mathbf{m}}_i = \frac{\sum_{j=1}^{N} \mu_{ij} \mathbf{x}_j}{\sum_{j=1}^{N} \mu_{ij}} \tag{2}$$

where μ_{ij} be the membership in the i'th class of the j'th labeled sample set.

The between-class fuzzy scatter matrix S_{FB} and within-class fuzzy scatter matrix S_{FW} are defined as follows, respectively.

$$S_{FB} = \sum_{i=1}^{c} N_i (\tilde{\mathbf{m}}_i - \overline{\mathbf{m}})(\tilde{\mathbf{m}}_i - \overline{\mathbf{m}})^T \tag{3}$$

$$S_{FW} = \sum_{i=1}^{c} \sum_{x_k \in C_i} (\mathbf{x}_k - \tilde{\mathbf{m}}_i)(\mathbf{x}_k - \tilde{\mathbf{m}}_i)^T = \sum_{i=1}^{c} S_{FW_i} \tag{4}$$

The optimal fuzzy projection W_{F-FLD} and the feature vector transformed by the fuzzy-based fisherface method can be calculated as follows.

$$W_{F-FLD} = \arg\max_{W} \frac{\left| W^T S_{FB} W \right|}{\left| W^T S_{FW} W \right|} \tag{5}$$

$$\tilde{\mathbf{v}}_i = W_{F-FLD}^T \mathbf{x}_i = W_{F-FLD}^T E^T (\mathbf{z}_i - \overline{\mathbf{z}}) \tag{6}$$

3 Online Signature Recognition System Using LDA and PPP Matching Technique

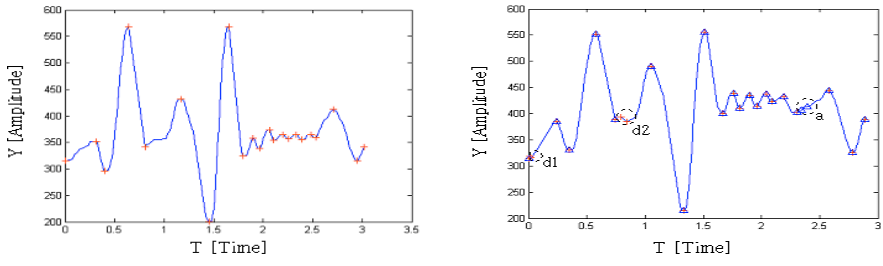
In a preprocessing, a signature data is resampled and normalized. Here, the number of resampled data is fixed at 80 for all data, and range of normalized data is between 0 and 1. The procedure of signature recognition system is as follows. First, we choose the feature data belonging to class (i) as shown in (7).

$$\text{Class (i)} = \sum_{k=1}^{80} \{x^i{}_k, y^i{}_k, t^i{}_k\}, \text{ for i} = 1 \text{ to n} \tag{7}$$

where, n is number of class, x and y are variation of X-axis and Y-axis respectively, and t is time. And then, we can get feature vectors for signatures from the conventional PCA and LDA method [1,2,5].

Now, we briefly describe a segment-segment matching method. First, we choose some peak points such as P1, P2, P3 in Fig. 1(a) and P1', P2', P3' in Fig 1(b). These points consistently exist regardless of the variation between two signatures. These points are referred to as PPP(Partitioning Peak Points). The PPP of a reference signature are usually consistent in most signatures, among them we select three highest values as PPP. The PPP of an input signature are selected by referring to the location of the sequence normalized based on the PPP of a reference signature.

After selecting the partition section of a comparison signature, perform the process of matching peak points and valley points of a reference signature and an input signature in each section. The matching is made by locating peak points and valley points of similar comparison signature with sequence location based on a reference signature after normalizing a reference signature. Under the condition that the sequence location, for the peak points and valley points of a reference signature, is compared with those of an input signature in each section, if the number of the two points is more than that of an input signature, the peak points or valley points of the input signature which has big difference in the sequence are deleted while ones which has small difference are added. In this case, it is highly possible that there is an error in addition/deletion of peak points and valley points if the sequence information is only used in matching. In order to reduce this error, the peak points and the valley points are respectively separated and then we extract correspondent points between reference signature and input signature. Fig. 1 shows the result which detected the correspondent points between reference signature and input signature. Here, "+" means peak points and valley points before matching while "ρ " means ones after matching. In addition, "d1"• and "d2 " mean where the unnecessary peak points and valley points are deleted while "a" means those are added.



(a) Reference signature (b) Input signature

Fig. 1. Matched peak and valley points between reference and input signatures

After the matching procedure, Euclidean distance is calculated between reference signature and input signature at peak and valley points. Here, we select time information which means duration between a peak and valley as feature vectors. The final decision step has a structure adding each error calculated by two methods.

4 Multi-modal Biometrics System Using New Decision Rule Based on SVM

The proposed multi-modal biometric system consists of a face recognition module, a signature recognition module, and a decision module as shown in Fig. 2. Here, the fuzzy LDA method is used for face recognition and the PPP matching method with the LDA is applied to the signature recognition system. As a final step, decision module is designed by the SVM which is capable of non-linear classification. The foundations of SVM have been developed by Vapnik and are gaining popularity due to many attractive features, and promising empirical performance [6].

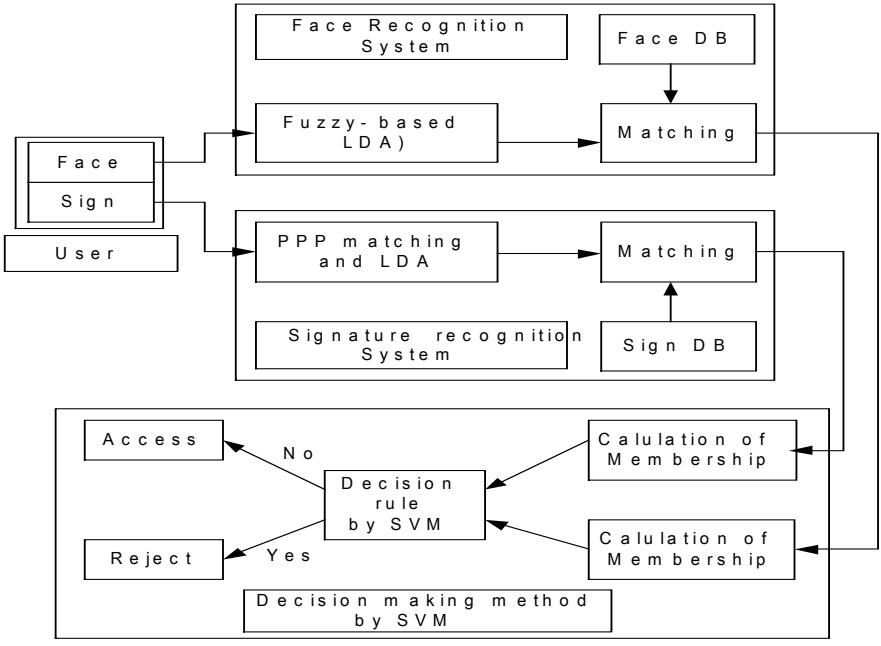


Fig. 2. Proposed multi-modal biometric system

To make a decision for access/reject, we compute each matching degree for face recognition and signature recognition. However, it is not preferable to use the matching values directly due to different ranges. Generally, distribution of matching values calculated by Euclidian distance between training data and test data between authorized person and imposter has the shape of Gaussian function. The normalization processes are as follows.

Let u_i and σ_i be the mean and the standard deviation of the matching values for an authorized person. The 95% of the matching values for true and imposter claim lie

in the $[\mu_i - 2\sigma_i, \mu_i + 2\sigma_i]$ and $[\mu_i + 2\sigma_i, \mu_i + 6\sigma_i]$, respectively [7]. Therefore, the original matching value $O_{i,orig}$ is mapped using a sigmoid as shown in Eq. (8).

$$O_i = \frac{1}{1 + \exp[(-\tau_i(O_{i,orig})]} \tag{8}$$

where,

$$\tau_i(O_{i,orig}) = \frac{O_{i,orig} - (\mu_i - 2\sigma_i)}{2\sigma_i} \tag{9}$$

Each membership value O_i obtained by Eq. (9) is used for input feature vectors of SVM to decide on access/reject. Then the SVM optimizes supporting patterns by maximizing the gap between the authentic patterns and the imposter patterns regardless of the data distribution. By using other kernel function, SVM can perform better classification. The kinds of kernel function are as follows.

Linear kernel function: $K(x, y) = w_1 \bullet x + w_2 \bullet y + b$

Polynominal kernel function: $K(x, y) = [x \bullet y) + 1]^d$

Gaussian Radial basis kernel function: $K(x, y) = \exp\left[-\frac{(x-y)^2}{2\sigma_2}\right]$

Exponential Radial basis kernel function: $K(x, y) = \exp\left[-\frac{|x-y|}{2\sigma_2}\right]$

B splines kernel function: $K(x, y) = B_{2n+1}(x - y)$

Sigmoid kernel function: $K(x, y) = \tanh(ax \bullet y + b)$

In case of selecting the best kernel function, there has not been a theoretical method but usually choose it by trial and error method.

5 Experiments and Analysis

5.1 Face Recognition Using the Fuzzy LDA Algorithm

First, we perform the face recognition for the established CNU (Chungbuk National University) face database. The CNU database contains 400 face images from 40 individuals in different situations. In the experiments, we use 200 face images from 20 individuals. The total number of images for each person is 10. They vary in face pose and light variation. The size of original image is 640×480. Each image was

resized as 112×92 pixel array whose gray level ranged between 0 and 255. Samples of the CNU face database are shown in Fig. 3.

The number of training and testing set are 5 images respectively. This procedure has been repeated for the ten times by randomly choosing different training and testing set. The 400 eigenvalue is obtained by PCA, here, we determined 40 eigenvectors representing the best performance in the ten times experiments. Also, the number of discriminant vectors is 9. Table 1 shows the comparison of mean and standard deviation for recognition rates in CNU database. As shown in the Table 1, the proposed method obtained a better recognition rates than previous ones.

Since PCA retains unwanted variations due to lighting and facial expression, the recognitions show a poor performance. We see that the fuzzy LDA method can be useful in uneven illumination.

Fig. 3. Samples of face image in the CNU face database

Table 1. Comparison of mean and standard deviation for recognition rate

Method Recognition	Eigenface (PCA)	Fisherface (PCA+LDA)	Fuzzy-based fisherface (Fuzzy+PCA+LDA)
Recognition rate	77±3.91%	94.8±3.29	96.8±1.68

5.2 Signature Recognition System Using LDA and PPP Matching Technique

We use Intuos 4×5 tablet from WACOM which takes about 100 points per second to construct the CNU signature database. The database contains 400 signature including 200 genuine signatures and 200 forgery signatures written 10 times by each one for 20 individuals. Samples of the CNU signature database are shown in Fig. 4.

The number of training and testing set are 5 signatures, respectively. This procedure has been repeated for the ten times by randomly choosing different training and testing set. Fig. 5 shows FAR (False Acceptance Rate) and FRR (False Reject Rate) according to methods such as PCA+LDA, PPP matching, and both. Here, FAR is defined as the rate of an imposter being accepted as a genuine individual and FRR is defined as the rate of a genuine individual being rejected as an imposter. In case of using PCA+LDA method, it shows better performance against random forgery signatures but poor performance against skilled forgery signature. On the other hand,

the method using PPP matching technique shows good performance against skilled forgery signatures. Finally, the fusion method using LDA, PCA and PPP matching method shows better performance against both random and skilled forgery signatures. Therefore, the proposed method is useful in a robust signature recognition system.

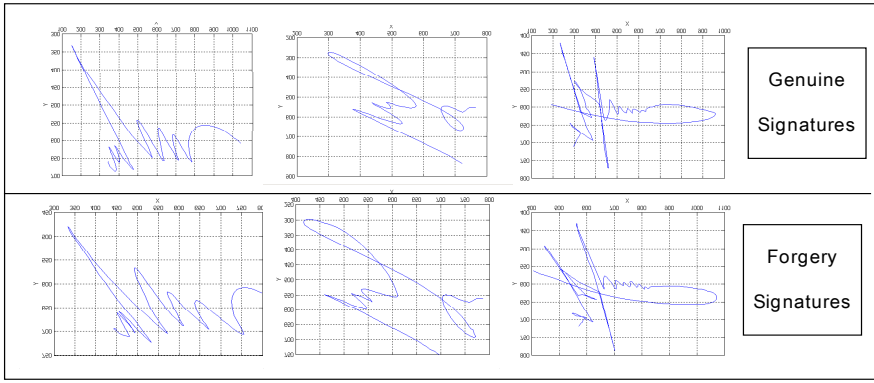
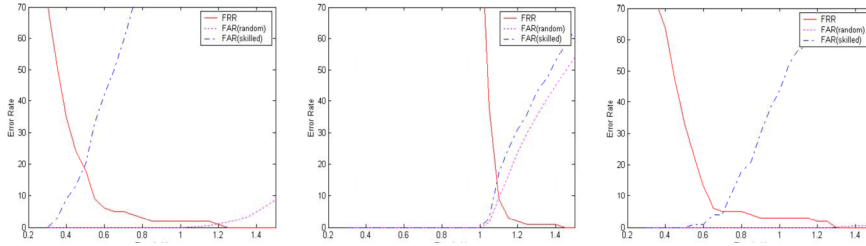


Fig. 4. Samples of signature in the CNU signature database



(a) PCA+LDA method (B) PPP matching method (C) Fusion method

Fig. 5. Comparison of FAR and FRR for various methods

5.3 Multi-modal Biometric System Using Support Vector Machine

To evaluate the proposed multi-modal biometric system, we use both the CNU face database and signature database. The number of training and testing set are 5 for 20 individuals, respectively. Test data is divided two classes such as evaluation data and verification data. Here, the evaluation data is used in designing an optimal hyperplain and verification data is used to verify the performance of the decision rule obtained in the evaluation step. The number of evaluation data and verification data are 2 and 3 respectively. In addition to these data, we use 200 databases for 20 individuals to verify the performance against an imposter with skilled forgery signature.

The proposed decision making rules are compared with weighting sum method [7], decision tree method [8], and fuzzy integral method [9]. We choose the FAR, FRR, and sum of FAR and FRR as the performance indices.

Table 2. Experiment result

Step / Method	Training step (%)			Test step (%)		
	FAR	FRR	FAR+FRR	FAR	FRR	FAR+FRR
Face	7.5	7.5	15.0	0	10.0	10.0
Signature	5	5	10.0	8.3	5.0	13.3
Weighted Sum Rule	2.5	2.5	5.0	5.0	3.33	8.33
Decision Tree	0	0	0	0	5.0	5.0
Fuzzy Integral	5.0	5	10.0	6.6	5.0	11.6
SVM (Linear)	5.0	2.5	7.5	0	1.66	1.66
SVM (Poly)	2.5	0	2.5	0	1.66	1.66
SVM (RBF)	5.0	2.5	7.5	0	0	0

From the various experiments, the proposed methods show better recognition rates than other ones as shown in Table 2. Specially, the SVM with RBF kernel function shows perfect authentic performance for testing data set.

6 Concluding Remarks

In this work, we suggested a multi-modal biometric scheme and evaluated its performance. Since the adopted fuzzy LDA method assigns the fuzzy membership value to the feature vector of a face image, it can reduce the sensitivity to similar variation between the face images due to illumination and pose. Simulation results show better recognition results than other ones such as eigenface and fisherface method. In case of signature recognition, LDA method shows better performance against random forgery signatures but poor performance against skilled forgery signatures. On the other hand, PPP matching method showed better performance against skilled forgery signatures but poor performance against random forgery signatures. The proposed robust online signature recognition method, however, has a good property of utilizing the complementary characteristics of two methods. Finally, we proposed more effective decision making method to combine two biometric systems. The method was designed by the support vector machine based on the probability distribution between authorized person and imposter. From the experimental results, we confirm that the proposed method can be applied to the applications of authentication where high performance is required.

Acknowledgements. This work was supported by grant No. R01-2002-000-00315-0 from the Basic Research Program of the Korea Science & Engineering Foundation.

References

[1] M. Turk and A. Pentland, "Eigenfaces for Recognition", Journal of Cognitive Neuroscience, Vol 3, pp. 72-86, 1991.

[2] Wenyi Zhao, Arvindh Krishnaswamy, Rama Chellappa," Discriminant Analysis of Principal Components for Face Recognition", Face Recognition from Theory to Application, Springer, 1998.

[3] Kiran G. V., Kunte R. S. R. and Saumel S., "On-line signature verification system using probabilistic feature modeling", Signal Processing and its Applications, Sixth International Symposium, Vol. 1, pp. 351-358, 2001.

[4] Ma Mingming, "Acoustic on-line signature verification based on multiple models", Computational Intelligence for Financial Engineering, (CIFEr) Proceedings of the IEEE/IAFE/INFORMS Conference, pp. 30-33, 2000.

[5] H. C. Kim, D. Kim, S. Y. Bang, Face recognition using the mixture-of-eigenface method, Pattern Recognition Letters, Vol.23, pp. 1549-1558, 2002.

[6] Vapnik. V.," The Nature of Statistical Learning Theory",Springer, 1995.

[7] Arun Ross, Anil Jain, "Information fusion in biometrics", Pattern Recognition Letters, Vol.24, pp. 2115-2125, 2003.

[8] Richard O. duda, Peter E. Hart, David G. Stock, "Pattern Classification", Second Edition, Wiley&Sons, Inc., 2001.

[9] Sung-Bae Cho, Jin H. Kim, "Multiple Network Fusion Using Fuzzy Logic", IEEE Trans. on Neural networks, Vol.6, No.2, 1995.

Using 3D Spatial Relationships for Image Retrieval by XML Annotation*

SooCheol Lee[1], EenJun Hwang[1], and YangKyoo Lee[2]

[1] Graduate School of Information and Communication
Ajou University, Suwon, Korea
[2] Department of Civil and Environmental Engineering
Daelim College, Anyang, Korea

Abstract. Retrieval of images from image databases using spatial relationship can be effectively performed through visual interface systems. In these systems, the representation of images with 2D strings, which is derived from symbolic projections, provides an efficient and natural way to construct image index and is also an ideal representation for the visual query. With this approach, retrieval issue is reduced to matching two symbolic strings. However, 2D-string representations might not specify spatial relationships between the objects in an image. Ambiguities arise for the retrieval of images of 3D scenes. In order to remove ambiguity in the description of objects' spatial relationships, in this paper, images are referred by considering spatial relationships using the spatial location algebra for the 3D image scene.

1 Introduction

The emergence of multimedia technologies and the possibility of sharing and distributing image data through large-bandwidth computer networks have emphasized the role of visual information. Due to the low cost of digital cameras, scanners, storage, transmission devices, and digital images are now currently employed in an eclectic range of different areas such as entertainment, art galleries, advertising, medicine, and geographic information systems, among others. Image retrieval systems support either high-level semantics-based retrieval, which defines image content at the conceptual level, or visual content based retrieval , which is based on the perceptual features like color, texture, structure, object shape, and spatial relationships.

Image retrieval based on spatial relationships among image objects is generally known as spatial similarity based retrieval. This type of retrieval has been identified as an important class of similarity-based retrievals and is used in the applications such as geographic and medical information systems.

* This work was supported by grant No. R05-2002-000-01224-0(2004) from the Basic Research Program of the Korea Science & Engineering Foundation.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 838–848, 2004.
© Springer-Verlag Berlin Heidelberg 2004

Spatial relationship is a fuzzy concept and is thus often dependent upon human interpretation. A spatial similarity function assesses the degree to which the spatial relationships in a database image conform to those in the query image. A spatial similarity algorithm provides a ranked ordering of database images with respect to a query image by applying the spatial similarity function between the query image and the database image.

The meta-data stored in the database may not be complete in the following sense. For example, it may contain relationships that object A is behind object B, and B is behind C, but not the relationship implied by the two. This incompleteness may be due to any of the following reasons. Firstly, existing image processing algorithms may not be able to recognize all objects and their relationships. For example, since images are two-dimensional objects of three-dimensional scenes, some of the spatial relationships in the missing dimension may not be detectable by the image processing algorithm. The missing objects and relationships may be introduced manually. In this process some of the implied relationships may be left out to save time. Secondly, the implied relationships may not be stored explicitly in order to save space. This saving in space will be advantageous in a distributed environment where the meta-data is stored at the user site with limited storage capacity, while the actual images are stored in remote sites.

In this paper, we discuss how to use 3D interface to query 2D images from the database not query but retrieve in the above sentence.. A prototype system is presented which employs a 3D interface with navigation and editing facilities.

The rest of the paper is organized as follows. Section 2 describes some of the related works. Section 3 presents image indexing and symbolic coding of spatial relationship. Section 4 explains the spatial similarity algorithm used. Section 5 explains our prototype system and its user interface. Section 6 describes some of the experimental results. Finally, Section 7 concludes the paper.

2 Related Works

So far, many CBIR systems and techniques have been reported. The QBIC system [18] at IBM allows an operator to specify various properties of a desired image including shape, color, texture and location. The system returns a selection of potential matches to those criteria, sorted by a score indicating the appropriateness of the match. Pentland et al. [2] presented another CBIR system that incorporates more sophisticated representation of texture and limited degree of automatic segmentation. Virage [3] and Chabot [17] also identify materials using low-level image properties. But, none of these systems considers spatial properties in a way that supports object queries.

Jagadish has proposed a shape similarity retrieval based on a two-dimensional rectilinear shape representation. Two shapes are similar if the error area is small when one shape is placed on the top of the other.

Lee et al. [16] propose a new domain-independent spatial similarity and annotation-based image retrieval system. Images are decomposed into multiple regions of interest containing objects, which are analyzed to annotate and extract their spatial relationships.

Tanimoto suggested the use of picture icons as picture indices, thus introducing the concept of iconic index. This concept has been given a theoretical framework, where abstraction operations are formalized to obtain various picture indices and to construct icons to facilitate the accessing of pictorial data.
Chang et al. [15] developed the concept of iconic indexing by introducing the 2D string representation of an image. Since then, the 2D string approach has been studied further. 2D-H string is an extension of 2D string. 2D-PIR graph can consider both directional and topological relationships between all possible spatial object pairs in an image. 2D string and 2D-H string can only represent directional relationships since they use only one symbol to represent relationships such as overlap, contain, inside, cover, covered by, equal. 2D-PIR graph manages interval relationships for x and y-axis and topological relationships between all spatial object pairs in a picture but too much storage space are needed to manage this graph.
Eakins [5] classified image queries into three levels that range from the highly concrete to the very abstract. Level 1, the lowest level, comprises retrieval by primitive features such as texture, color, and shape. The system that corresponds to this is the QBIC and MIT PhotoBook [2]. Level 2 comprises retrieval by derived attributes involving some degree of logical inference about the identity of the objects depicted in the image. This level of query is of more general applicability than level 1. Level 3 comprises retrieval by abstract attributes, involving a high degree of abstract and possibly subjective reasoning about the meaning and purpose of the object or scene depicted.

3 Image Indexing for Image Retrieval in Database

One of the most important issues in the design of image database system is the representation of image contents that allows for efficient storage and retrieval of image data through a user-friendly interface. An image contains two types of information: information regarding its objects and information related to the shape and the spatial arrangement of its image elements. In order to make an image database flexible, this spatial knowledge should be preserved by the data structure that is used to store the images.
Image contents can be described in terms either of the image objects and their spatial relationships or of the objects in the original scene and their spatial relationships. In the first case, 2D objects are involved, and 2D spatial relationships are evaluated directly on the image plane. In the second case, scenes associated with images involve objects that differ greatly in their structural properties from one application to the other. Specifically, scenes involve objects if objects have prevalently a 2D structure or involve 3D objects if they are common real-world scenes. Spatial relationships for the two cases are 2D and 3 D, respectively.
ROIs(Region of Interest) [12, 16]for querying image databases are special images defined through a computer system by the user. They maintain a symbolic relationship with objects in the real world. According to the type of application, either 2D or 3D structure can be associated with each icon to build virtual scenes with 2D or 3D objects, respectively.
Following the Query by Pictorial Example(QPE) [14] approach, the index of an image is an iconic image itself which represents the visual information contained in the

image in a form suitable for different levels of abstraction and management. In particular, the QPE philosophy expresses the objects and the spatial relations to be retrieved through a symbolic image which serves as a query and is matched against the images in the database.

3.1 Symbolic Coding of Spatial Relationships

Every image in the database contains a set of unique and characterizing image objects that scatter in any arbitrary locations. There could exist various spatial relationships among these image objects. The spatial location can be represented either by the relative coordinate or by the absolute coordinate.

In a 3-D space, the spatial location of an object O in an image is represented as a point P_o where $P_o = (X_o, Y_o, Z_o)$, and an image itself as a set of points $P=\{P_1, P_2,\ldots, P_n\}$, where n is the number of objects of interest in the image. These points are tagged or annotated with labels to capture any necessary semantic information of the object. We call each of these individual points representing the spatial location of an image object a *spatial location point*[15]. For the sake of simplicity, we assume that the spatial location of an image object is represented by only single spatial location point, and hence, the entire image is represented by a set of spatial location points. In order to represent spatial relations among the spatial location points of an image, we decompose an image into four equal size quadrants. Figure 1 shows an image whose three spatial location points are located at the different quadrants.

This representation scheme is translation, orientation and scale independent. Based on this scheme, we can define image location algebra for image objects X, Y and Z as shown in Table 1.

Suppose we are given an image with *n* objects of interest. We can describe its spatial information using unidirectional graph where each object corresponds to a vertex and any spatial relationship between two objects is indicated by a label along their edge. We refer to this graph as *spatial graph*. Figure 2 shows an original image and its spatial graph for the image objects.

Table 1. Image location algebra

Notation	Operator	Meaning
X < Y	Lupper	X is located in Left upper of Y
X ∧ Y	Llower	X is located in Left lower of Y
X > Y	Rupper	X is located in Right upper of Y
X ∨ Y	Rlower	X is located in Right lower of Y
X ∪ Y	Upper	X is located in Upper of Y
X ∩Y	Below	X is located in Below of Y
X]Y	Right	X is located in Right of Y
X [Y	Left	X is located in Left of Y
X / Y	Center	X or Y is located in Center of M
X % Y	Overlap	X is Overlapped of Y
X ⊗ Y	Inside	X is Inside of Y
X ⊕ Y	Outside	X is Outside of Y
X • Y	In front of	X is In front of Y

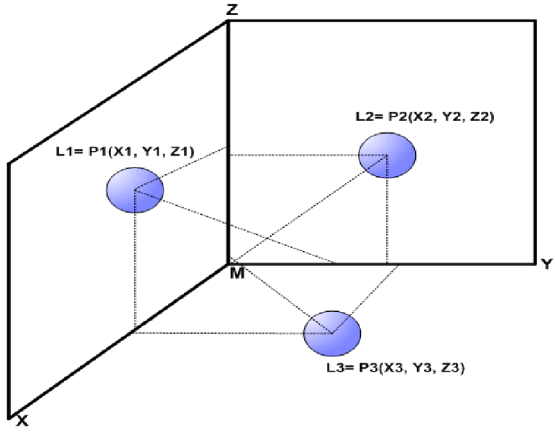


Fig. 1. Spatial location point

***Definition* 1**. A spatial graph is a set of pair (V, E) where:

$V = \{L1, L2, L3, \ldots, Ln\}$ is a set of nodes, representing objects.
$E = \{e1, e2, e3, \ldots, en\}$ is as set of edges, where each edge is connecting two nodes L1 and L2, and labeled with a spatial relationship between them.

In the figure, some of the spatial relationships among the spatial location points are represented along their edges using image location algebra. Here, M is a special spatial point called fiducial point. their edges using image location algebra.

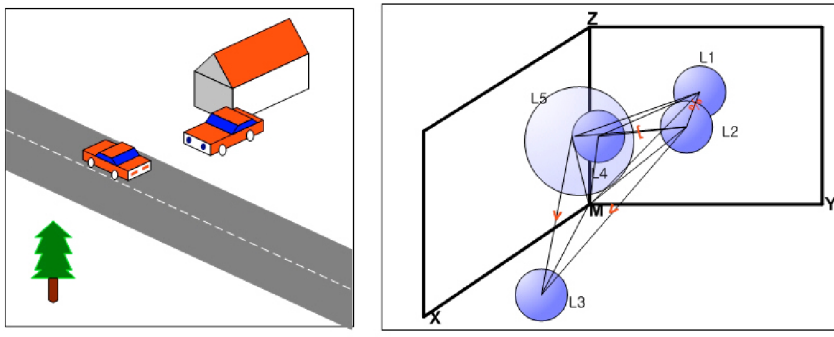


Fig. 2. Original image and its spatial graph

According to the definition 1, the spatial relationships for the figure are:

$V = \{L1, L2, L3, L4, L5\}$
$Rel = \{L1 \wedge M\}, \{L1 \,\%\, L2\}, \{L1 \wedge L3\}, \{L1 \wedge L4\}, \{L1 \wedge L5\}, \{L2 \wedge M\}, \{L2 \,\%\, L1\}, \{L2 \wedge L3\}, \{L2\ [L4\}, \{L2\ [L5\}, \{L3 < M\}, \{L3 > L1\}, \{L3 > L2\}, \{L3$

> L4}, {L3 > L5}, {L4 ∩ L1}, {L4] L1}, {L4 ∧ L2}, {L4 ∧ L3}, {L4 % L5}, {L5 ∪ L1}, {L5 [L2}, {L5 ∧ L3}, {L5 ⊗ L4}

4 Metric for Measuring Spatial Similarity an Algorithm

The crucial point in similarity-based retrieval is to determine similarity metric that is efficient to calculate and capture the essential aspects of similarity that humans recognize. By quantifying the concept of "degree of similarity", we can measure the similarity degree between an image and a query image in terms of objects and their spatial relationships. In this paper, we denote the degree of similarity as a value in the range [0, 1]. For example, the degree of similarity between image P1 with object L1, L2, L3 and P2 with L4, L5, L6 is dependent on the degree of similarity between L1 and L4, L2 and L5, L3 and L6, respectively.

In this paper, we introduce the operator neighborhood graph[16] which formally defines the distances among the spatial operators.

***Definition* 2.** Spatial relationships between two objects are neighboring if they can be directly transformed to each other by a deforming operations (scaling, moving, and rotating).

Figure 3 shows an operator neighborhood graph for the spatial relationships corresponding to the spatial operators in Table 1. The distance between two spatial operators δ_1 and δ_2 is defined by the shortest path from δ_1 to δ_2 on the neighborhood graph and is denoted by *distance* (δ_1, δ_2). The maximum distance on the neighborhood graph is 4 and the minimum is 0. We define the similarity degree using the following formula:

$$Sim_Obj(\delta_1, \delta_2) = 1 - (distance((\delta_1, \delta_2)/D_{max})$$

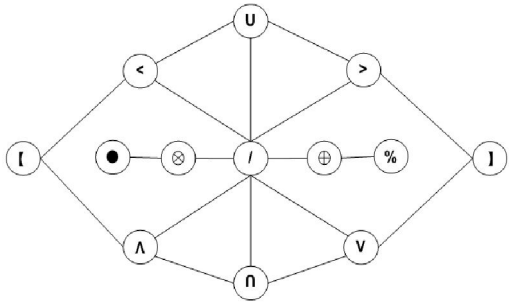


Fig. 3. Operator neighborhood graph

Table 2 shows the similarity degree that is derived from the operator neighborhood graph. Using the table, we can measure the similarity between two images.

Table 2. Similarity between spatial operators

SIM	<	∧	>	∨	∪	∩	]	[	/	%	⊗	⊕	•
<	1	0.5	0.5	0.5	0.5	0.5	0.25	0.75	0.75	0.25	0.5	0.5	0.25
∧	0.5	1	0.5	0.5	0.5	0.75	0.25	0.75	0.75	0.25	0.5	0.5	0.25
>	0.5	0.5	1	0.5	0.75	0.5	0.75	0.25	0.75	0.25	0.5	0.5	0.25
∨	0.5	0.5	0.5	1	0.5	0.75	0.75	0.25	0.75	0.25	0.5	0.5	0.25
∪	0.5	0.5	0.75	0.5	1	0.5	0.5	0.5	0.75	0.25	0.5	0.5	0.25
∩	0.5	0.75	0.5	0.75	0.5	1	0.5	0.5	0.75	0.25	0.5	0.5	0.25
]	0.25	0.25	0.75	0.75	0.5	0.5	1	0	0.5	0	0.25	0.25	0
[	0.75	0.75	0.25	0.25	0.5	0.5	0	1	0.5	0	0.25	0.25	0
/	0.75	0.75	0.75	0.75	0.75	0.75	0.5	0.5	1	0.5	0.75	0.75	0.5
%	0.25	0.25	0.25	0.25	0.25	0.25	0	0	0.5	1	0.25	0.75	0
⊗	0.5	0.5	0.5	0.5	0.5	0.5	0.25	0.25	0.75	0.25	1	0.5	0.75
⊕	0.5	0.5	0.5	0.5	0.5	0.5	0.25	0.25	0.75	0.75	0.5	1	0.25
•	0.25	0.25	0.25	0.25	0.25	0.25	0	0	0.5	0	0.75	0.25	1

Query result is a set of images satisfying the condition expressed in it, i.e., the images than have a similarity degree greater that a specified threshold with respect to the query. We can define user query as follows:

***Definition* 3.** User query Q is a triple (ξ, S, t) where ξ is the set of objects in the query image. S is the set of spatial relationships among objects and t is the minimal required similarity between Q and database image (I_{DB}) and is between 0 and 1, inclusively.

As an example, let us consider the image in Figure 2. According to the definition 3, we can define the query image as follow:

$Q = \{\xi, S, t\}$

$\xi = \{L1, L2, L3, L4, L5\}$

S = {(L1 ∧ M), (L1 % L2), (L1 ∧ L3), (L1 ∧ L4), (L1 ∧ L5), (L2 ∧ L3), (L2 [L4), (L 2 [L5), (L3 > L1), (L3 > L4), (L3 > L5), (L4 ∩ M),(L4 ∧ L3), (L4 % L5), (L5 ⊗ L4)}

Given a query image Q of n objects, we can calculate its spatial similarity to a database image I_{DB} by the following function.

$$Sim_Deg\ (Q, I_{DB}) = \sum_{i=1}^{n} \sum_{j=1}^{n-1} \left| O_{i}^{q}\, \vartheta\, O_{j}^{I_{DB}} \right|$$

In the formula, O_{1}^{q} and $O_{1}^{I_{DB}}$ are spatial relationships between the query and database images. Symbol ϑ compares two spatial operators by the value of similarity in Table 2. According to the formulation, the spatial similarity value between O_{1}^{q} = L1 % L2 and $O_{1}^{I_{DB}}$ = L5 ⊗ L4 is 0.25.

The *Sim_Deg* function has been effectively used as a metric for measuring image similarity in our prototype content-based retrieval system.

5 Implementation

We have implemented a prototype system based on three basic principles:

To describe real world images in the database in terms of the 3D description of the scene that they represent;

To query through virtual scene that is defined with a direct manipulation of 3D symbol;

To represent spatial relationships between objects both in the original and virtual scenes according to the representation language expounded previously.

Images and their descriptions are stored in an XML database. XML(Extensible Markup Language) is a simple, very flexible text format derived from SGML. Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in the exchange of a wide variety of data on the Web and elsewhere.

XML permits document authors to create markup for virtually any type of information. This extensibility enables document authors to create entirely new markup languages for describing specific types of data, including mathematical formula, chemical molecular structures, music, recipes, etc.

We have represented spatial relationships and features of image objects using XML.

Figure 4 shows a snapshot of implemented image analysis process. For each image object marked by a rectangle, its edge is extracted and any necessary semantic information is coded into an XML document. The window on the left bottom shows the spatial graph for the marked image objects. Image objects in spatial graph can be modify its 3D location using forward and back button.

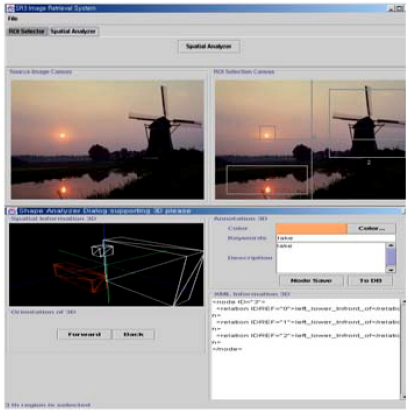


Fig. 4. Image analyzer interface

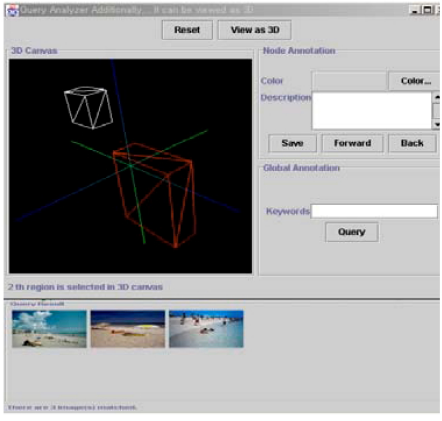


Fig. 5. Query window interface

These represent object location and instance. The visual interface of the system is comprised of a ROI interface which supports the selection of images from the

database and several operating facilities, as well as Query window, which is used for the definition and the visualization of the 3D query as shown in figure 5.

Table 3. Query Q1 result

#	Spatial Constraint	Color	Keywords	Retrieved	Relevant	Precision
1	"-"		Tree	215	5	2.3
2	"-"	Red	House	47	5	10.6
3	"-"	Red	House and Tree	21	5	23.8
4	Left or Right	Red	House and Tree	13	5	38.5
5	In front of	Red	House and Tree	5	5	100

Table 4. Query Q2 result

#	Spatial Constraint	Color	Keywords	Retrieved	Relevant	Precision
1	"-"	"-"	Car	865	24	2.8
2	"-"	White	Car	589	24	4.0
3	"-"	White	Car and Garage	430	24	5.6
4	"-"	White	Car, Garage and Lake	142	24	17
5	Left or Right	White	Car, Garage and Lake	57	24	42.1
6	Overlap or Left or Right	White	Car, Garage and Lake	43	24	56
7	Overlap or Inside	White	Car, Garage and Lake	18	7	39
8	Inside or Left or Right	White	Car, Garage and Lake	24	24	100

6 Experimental Results

Using the image retrieval system, we have tested some of the typical queries under different combinations of stored annotation and spatial constraint of image objects. The image database contains 2,000 commercial Corel images that cover a wide range of nature scenes, buildings, construction sites, animals, etc. Table 3 shows some of the results for a query would be:

Q1: "Find the images with a red house in front of the tree"

Q2: "Find the white car images inside the garage which is beside the lake"

In the query Q1, retrieve all house images in the database that were in front of the tree and have a red color. There are 5 images in the collection that satisfy the query.

In the table 3, queries 1-3 used keywords search only, query 4 used a combination of color, keywords and 2D spatial constraint and query 5 used combination of color, keywords and 3D spatial constraint.
Using the keywords "house and tree" in conjunction with the spatial constraint "in front of" gives best precision.
The query Q2 returns all white car images in the database that were beside the lake and inside the garage. There were 24 images in the collection that satisfy the query.
Experiments have revealed that retrieving images based on keywords only gives marginal results. However, when incorporating spatial constraint in the query gives much better result.

7 Conclusion

In this paper, the subject of retrieval of contents of images depicting 3D scenes through 3D interface has been addressed. In this approach, images are referred to by spatial relationships between objects in the 3D image scene.
We have built a prototype system based on reduction rules and performed experiments to see how it works for typical image retrieval requests. The prototype system provides tools for analyzing, annotating, querying, and browsing images in a user friendly way.

References

1. A. Gupta and R. Jain, "Visual information retrieval," *Comm. Assoc. Comp. Mach.*, May 1997.
2. A. Pentland, R. Picard and S. Sclaroff, "Photobook: Content-based manipulation of image databases," *SPIE Proc. Storage of Retrieval for Image and Video Databases*, February 1994.
3. C. Carson, S. Belongies, H. Greenspan and J. Malik, "Region-based image querying," *Proc. IEEE Workshop on Content-Based Access of Image and Video Libraries*, June 1997.
4. Chang, C. C. and Lee, S. Y, "Retrieval of similar pictures on pictorial databases," Pattern Recognit., Vol 24-7, pp675-681, 1991.
5. Eakins, J. P., " Automatic image content retrieval: Are we going anywhere?," *In proceedings of the 3rd Int. Conf. on Electronic Library and Visual information Research.*, May. 1996.
6. Egenhofer M. J. and Franzasa R. D, "Point set topological spatial relations., *Journal Geogr. Information System.,* vol 5-2, pp 161-174, 1991.
7. Gudivada, V. N. and V.V. Raghavan, "Design and evaluation of algorithms for image retrieval by spatial similarity," *ACM Trans. on Information Systems*, 13(2), 1995.
8. Gudivada, V. N and Jung, G. S, "An algorithm for content-based retrieval in multimedia databases," *Proc. Int. Conf. on Multimedia Computing and Systems*," pp90-97, 1995.
9. J. R. Smith and S. –Fu Chang, "Visual SEEK: A Fully Automated Content Based Image Query System," *Proc. ACM Mult. Conf.*, *Boston Ma.*, *Nov. 1996*.

10. J. R. Smith and S. –Fu Chang, "Tools and techniques for color image retrieval," *Proc. IEEE Int. Conf. on Image Proc.*, pp 52-531, 1995.
11. J.Huang, S.R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih, "Image indexing using color correlograms," *Proc. IEEE Comp. Soc. Conf. Vis. and Patt.*, pages 762-768, 1997.
12. L. H. Rodrigues, Building Imaging Applications with Java Technology, *Addison Wesley*, 2001.
13. Lee, S.Y. and F.J. Hsu, "Spatial reasoning and similarity retrieval of images using 2D-C String knowledge representation," *Pattern Recognition*, Vol 25-3, pp. 305-318, 1992.
14. M. Nabil, A.H.H. Ngu, and J. Shepherd, "Picture Similairty Retrieval Using the 2D Projection Interval Representation," *IEEE Trans. Knowledge and Data Eng.*, vol. 8, no. 4, pp. 533-539, Aug. 1996.
15. S. Chang, Q. Shi and S. Yan, "Iconic indexing using 2-D strings," *IEEE Trans. on Pattern Analysis & Machine Intelligence*, Vol. 9, No. 3, pp. 413-428, 1987.
16. S. Lee and E. Hwang, "Spatial Similarity and Annotation-Based Image Retrieval System", *IEEE Fourth International Symposium on Multimedia Software Engineering*, Newport Beach, CA, December 2002.
17. V. E. Ogle and M. Stonebraker, "Chabot: Retrieval from a Relational Database of Images," *IEEE Computer*, Vol. 28, No. 9, September 1995.
18. W. Niblack, et al. "The QBIC project: Query images by content using color, texture and shape," *SPIE V 1908*, 1993.

Association Inlining for Mapping XML DTDs to Relational Tables*

Byung-Joo Shin and Min Jin

Department of Computer Science & Engineering, Kyungnam University, Masan, KOREA
challenger@hawk.kyungnam.ac.kr, mjin@kyungnam.ac.kr

Abstract. In this paper, we propose a new inlining method called the *Association inlining* for mapping DTDs to relational tables. It extends *Shared inlining* and *Hybrid inlining* to reduce relational fragments and excessive joins. In conjunction with the *Association inlining*, a *Path table* that contains relational schema information of the path from the root to every element is provided. The schema information extracted from the path table is exploited in processing XML queries. The performance of our method is compared to those of *Shared* and *Hybrid inlining* methods in terms of the number of joins and queries. The experiments shows that the number of joins of our method is less than that of *Shared* and the number of subqueries per query is less than that of *Hybrid* in general.

1 Introduction

XML is becoming a de facto standard for exchanging data in Internet data processing environments due to the inherent characteristics such as hierarchical self-describing structures. Hence, the volume of XML documents is getting larger. This has given rise to the need of database technologies in storing and querying XML documents. There are two main approaches for storing voluminous XML documents. One approach is to develop special purpose XML repositories that support the XML data model and query languages directly. The other approach is to take advantage of existing database facilities by using conventional relational or object-oriented database systems. Since relational database systems are widely used and provide mature inherent services to be exploited in managing XML data, they are promising alternatives for storing XML documents[1,2,3,4,14].

However, data are represented in flat structures in relational databases, whereas XML documents are represented in hierarchical structures with nests and recursions. Hence, additional processing is required for storing and querying XML data in relational databases due to the structural discrepancy between XML and relational databases[7,8,9,15,16].

* This work was supported by Kyungnam University Research Fund.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 849–858, 2004.
© Springer-Verlag Berlin Heidelberg 2004

To cope with the problems, in this paper we propose an *Association inlining* for mapping XML DTDs to relational tables. Our approach combines the join reduction properties of *Hybrid inlining* with the sharing features of *Shared inlining*[10,11]. In addition to it, we store the schema information by using a *Path table*, which contains relational schema information of the path from the root to every element[13]. It is exploited in processing queries over XML data.

The rest of this paper is organized as follows. Section 2 briefly overviews the related work concerning storing XML data using relational databases. Section 3 describes how to store XML documents with DTDs using relational databases including the *Association inlining* method. Section 4 shows the performance of our method compared to those of *Shared* and *Hybrid* methods in terms of the number of joins and the number of subqueries per query. Section 5 offers conclusions.

2 Related Work

Methods for storing XML documents in relational databases can roughly be classified into two categories: *Model mapping approach* and *Structure mapping approach*[4,11,13]. *Model mapping approach* deals with storing XML documents without structural information. Relational schemas are defined regardless of the structural information of an XML document.

In contrast to *Model mapping approach*, the *Structure mapping approach* deals with storing XML documents with the structural information such as a DTD or a schema. Relational tables are generated based on the structural information extracted from the DTD or the XML schema. There have been three techniques called *Basic inlining*, *Shared inlining*, and *Hybrid inlining*. A relation is created for each element in *Basic inlining*. The principal idea behind *Shared* is to share the element nodes represented in multiple relations in *Basic* by creating separate relations for these elements. In *Shared*, relations are created for all elements having an in-degree greater than one in DTD graph. Nodes with an in-degree of one are inlined in the parent node's relation. Element nodes having an in-degree of zero are also made into separate relations. As in *Basic*, set sub-elements are made into separate relations. Of the mutually recursive elements all having in-degree one, one of them is made a separate relation. Representing element nodes in exactly one relation in *Shared* creates a small number of relations compared to *Basic*. *Shared* addressed some shortcomings of *Basic*, however, *Shared* performs worse than *Basic* in one respect, increasing the number of joins. This method is known to outperform other methods in data representation and performance with different datasets and different queries[6]. *Hybrid* is the same as *Shared* except that it inlines elements with in-degree greater than one that are not recursive elements or set sub-elements. Although this *Hybrid*'s property might lead to reduction of the number of joins per SQL query, it also causes more SQL queries to be generated[5,10,11,13].

3 Generation of Relational Schemas from DTDs

3.1 Association Inlining Approach

In this section, we describe how to generate relational schemas from XML DTDs to store XML documents in relational databases. Our approach combines the join reduction properties of *Hybrid inlining* with the sharing features of *Shared inlining*[10]. We call our approach an *Association inlining*.

3.1.1 Creating an XML DTD Graph

We create a DTD graph representing the structure of a DTD. We define a DTD graph as follows.

Definition 1. A directed XML DTD graph $G = (V, L)$ is an ordered pair of finite sets V and L. The elements of V are called nodes, $V = E \cup A$ where E is a set of elements and A is a multiset of attributes. The elements of L are called edges. Each edge joins two nodes e_i and e_j, which denoted (e_i, e_j) and it's type is one of the elements of the set called edge-types = {->, ?, +, *}.

Definition 2. For an element e, $e \in E$ and an edge l, $l \in L$

- *e.in-degree* is the number of edges that are incident to *e*.
- *e.out-degree* is the number of edges that are incident from *e*.
- *l.edge-type* is the type of edge *l*.
- *e.edge-type* is a set of *l.edge-type*, where *l* is incident to *e*.
- *e.children* is a set of elements to which an edge is incident from *e*.
- *e.descendants* is a set of elements that have a path from *e*.

```
<!ELEMENT publication (paper*, book*)>
<!ELEMENT paper (year, title, authors, reference)>
<!ATTLIST paper paperID ID #REQUIRED>
<!ELEMENT reference (paper*)>
<!ELEMENT book (year, title, authors)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT authors (author+)>
<!ELEMENT author (name, email?, address?)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT email (#PCDATA)>
<!ELEMENT address (#PCDATA)>
```

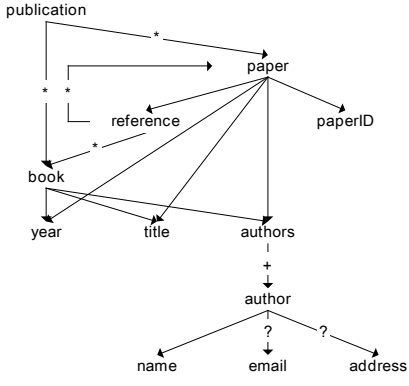


Fig. 1. A DTD specification and the corresponding DTD graph

A DTD and the corresponding DTD graph are given in Fig. 1.

3.1.2 Mapping a DTD Graph to Relational Tables

In this section, we describe how to map a DTD graph to relational tables. First, we define the some notions of elements used to map a DTD graph to relational schemas as follows.

Definition 3. An element e is an element-only element if and only if there is at least one edge (e, f) where $f \in E$ and no edges (e, a) where $a \in A$.

Definition 4. An element e is an empty element if and only if there is no edges (e, f) where $f \in E$ and there exists at least one edge (e, a) where $a \in A$.

Definition 5. The sequence of nodes, $P = e_1, e_2, e_3, \ldots, e_{(n-1)}, e_n$, where e_i is an element for $i = 1,2,\ldots,n-1$ and e_n might be either an attribute or an element, is a directed path from e_1 to e_n if and only if $(e_i, e_{i+1}) \in E$, $1 \geq i \geq n-1$. P is a simple directed path if and only if all nodes(except possible the first and last) are distinct. P is a directed cycle if and only if it is a simple directed path and $e_1 = e_n$.

Definition 6. There is a root node in the graph corresponding to the root element in the DTD. A node e_j is reachable from e_i if there is a simple directed path from e_i to e_j. The length of the path is the number of edges on the path.

- *e.length* is the number of edges on the shortest path from the root node to *e*.

Definition 7. When there is a directed cycle, $e_1, e_2, e_3, \ldots, e_{(n-1)}, e_n$ where $e_1 = e_n$, in a graph G, the node e_i such that $e_i.length$ is the smallest among $e_j.length$ for $j = 1,2, \ldots,n-1$. The edge that is incident to e_i is a recursive edge.

Definition 8. An element *e* is separable if *e.in-degree* = 0.

Definition 9. An element *e* is inlinable if the followings are satisfied;

- e.in-degree =1
- none of e.edge-type is contained in {+, ∗}.
- no recursive edges incident to e

We define the rules for mapping a DTD graph to relational tables as follows.

Rule 1. First of all, an element-only element that is not mapped to a table is eliminated in the mapping process. For an element-only element e,

i) If it is a root element and all of its children are mapped to separate tables, it is eliminated.

ii) If it is inlinable(it satisfies the conditions of Definition 9), it is eliminated.

iii) If *e.in-degree* ≥ 2, *e.out-degree* = 1, and the following conditions are satisfied, then it is eliminated.

- none of *e.edge-type* is contained in {+, ∗}.
- no recursive edges incident to *e*

Rule 2. If an element is separable and it is not element-only element, it is mapped to a separate table.

Rule 3. An element-only element that is a root element is mapped to a separate table unless at least one of its children is mapped to a separate table.

Rule 4. For an element *e*, if any of *e.edge-type* is contained in {+, ∗} or there is a recursive edge incident to *e*, it is mapped to a separate table.

Rule 5. For an element *e*, if *e.in-degree* ≥ 2 and any of *e.edge-type* is not contained in {+, ∗}, apply the following;

- For any element *f* where *f*∈ *e.descendants*, if *f.out-degree* ≥ 2 or *e.out-degree* ≥ 2, *e* is mapped to a separate table.

Rule 6. For an empty element *e*, if *e.in-degree* = 1, then it is not represented as an attribute of the parent element, instead the attributes of *e* are represented as attributes of the parent element of *e* directly.

3.1.3 An Example

First, we eliminate element-only elements that are not mapped to tables. Fig. 2 shows the DTD graph that is resulted from application of Rule 1 to the DTD graph in Fig. 1. The element *publication*, *reference* and *authors* are eliminated. Element *publication* is a root element and all of its children(element *paper* and *book*) are mapped to separate tables. Thus, it is eliminated by Rule 1.i). There is a directed cycle between *paper* and *reference* and *paper.length = 1*, *reference.length = 2*. The edge that is incident to *paper* is a recursive edge but the edge that is incident to *reference* is not a recursive edge by Definition 7. Therefore, element *reference* is regarded as inlinable and eliminated by Rule 1.ii). Although *authors.in-degree* = 2, element *authors* is eliminated by Rule 1.iii) since *authors.out-degree* = 1, none of *author.edge-type* is contained in {+, ∗}, and no recursive edges incident to it.

Next, we map the DTD graph in which element-only elements are eliminated to relational tables. Rule 2,3,4 and 5 are applied in the mapping process. Hence, element *paper*, *book* and *author* in Fig. 2 are mapped to separate tables. Though *year.in-degree* = 2 and *title.in-degree* = 2, they are mapped to attributes of the two parent elements(*paper* and *book*) instead of being mapped to separate tables due to the fact that *year.out-degree* = 0 and *title.out-degree* = 0.

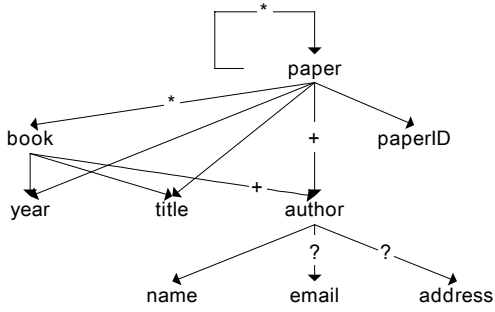


Fig. 2. The graph after elimination of element-only elements over the DTD graph in Fig. 1

The relational tables that were generated for the DTD graph in Fig. 1 and the XML document in Fig. 3 are shown in Fig. 4. Each table has an *ID* field that serves as the key of the table. All tables corresponding to elements having a parent also have a *parentID* field that serves as a foreign key. The *parentCode* field represents the corresponding parent table among multiple parent tables. The *order* field represents

the order of occurrence within the element. The *nested* field indicates the degree of recursions on the *parentCode* table. The *docID* field denotes the XML document.

```
<publication>
  <paper paperID="1">
    <year>2003</year>
    <title>Associaion Inlining…</title>
    <authors>
      <author>
        <name>Byung-Joo Shin</name>
        <email>raniman@zeus.kyungnam.ac.kr</email>
      </author>
      <author>
        <name>Min Jin</name>
        <email>mjin@kyungnam.ac.kr</email>
      </author>
    </authors>
    <reference>
      <paper paperID="2">
        <year>1999</year>
        <title>Relational Databases…</title>
        <authors>
          <author>
            <name>Shanmugasundaram.J</name>
          </author>
        </authors>
      </paper>
      <book>
        <year>2000</year>
        <title>Professional XML…</title>
        <authors>
          <author>
            <name>Williams.K</name>
          </author>
        </authors>
      </book>
    </reference>
  <book>
    <year>2002</year>
    <title>C++ XML</title>
    <authors>
      <author>
        <name>Arciniegas.F</name>
      </author>
    </authors>
  </book>
                    :
</publication>
```

Fig. 3. An XML document

Paper

ID	parentID	parentCode	order	year	title	nested	docID
1	NULL	NULL	1	2003	Association Inlining…	0	1
2	1	1	1	1999	Relational Databases…	1	1
:	:	:	:	:	:	:	:

Book

ID	parentID	parentCode	order	year	title	nested	docID
1	1	1	1	2000	Professional XML…	NULL	1
2	NULL	NULL	1	2002	C++ XML	NULL	1
:	:	:	:	:	:	:	:

Author

ID	parentID	parentCode	order	name	email	address	nested	docID
1	1	1	1	Byung-Joo Shin	raniman@zeus….	NULL	NULL	1
2	1	1	2	Min Jin	mjin@kyungnam….	NULL	NULL	1
3	2	1	1	Shanmugas…	NULL	NULL	NULL	1
4	1	2	1	Williams.K	NULL	NULL	NULL	1
5	2	2	1	Arciniegas.F	NULL	NULL	NULL	1
:	:	:	:	:	:	:	:	:

Fig. 4. Tables for storing the XML document in Fig. 3

3.2 Path Table

From the DTD graph that is not simplified in the mapping process, we could get a tree by eliminating recursive edges. The node that corresponds to a root element is

designated as a root of the tree. For every node in the tree, we could get a path expression from the root to it[11,13]. For the recursive expressions, the information will be added when the number of iterations on the XML documents is released. For each path expression, we represent the information of it using the following table that is called *Path table*. Fig. 5 shows the *Path table* of the DTD graph in Fig. 1. The delimiter '#' is added to use *Like* clause of SQL. *Table* and *column* mean the relational table and column that correspond to the last node on the *pathExp*. For an element *e* that is the last element in the path expression, *e.table* means the table that corresponds to the element *e*. For a node *e* that is the last in the path expression, *e.column* means the column that corresponds to the node in the table. *ParentCode* means the parent table on the *pathExp* in the hierarchy of relational tables.

pathID	pathExp	table	column	parentCode	recursions
1	#/publication	NULL	NULL	NULL	NULL
2	#/publication#/paper	paper	NULL	NULL	NULL
3	#/publication#/paper#/@paperID	paper	ID	NULL	NULL
4	#/publication#/paper#/year	paper	year	NULL	NULL
5	#/publication#/paper#/title	paper	title	NULL	NULL
6	#/publication#/paper#/authors	NULL	NULL	NULL	NULL
7	#/publication#/paper#/authors#/author	author	NULL	1	NULL
8	#/publication#/paper#/authors#/author#/name	author	name	1	NULL
9	#/publication#/paper#/authors#/author#/email	author	email	1	NULL
10	#/publication#/paper#/authors#/author#/address	author	address	1	NULL
11	#/publication#/paper#/reference	NULL	NULL	NULL	NULL
12	#/publication#/paper#/reference#/paper	paper	NULL	1	1
13	#/publication#/paper#/reference#/paper#/@paperID	paper	ID	1	1
:	:	:	:	:	:
21	#/publication#/paper#/reference#/book	book	NULL	1	NULL
22	#/publication#/paper#/reference#/book#/year	book	year	1	NULL
:	:	:	:	:	:
25	#/publication#/paper#/reference#/book#/authors#/author	author	NULL	2	NULL
26	#/publication#/paper#/reference#/book#/authors#/author#/name	author	name	2	NULL
:	:	:	:	:	:
29	#/publication#/book	book	NULL	NULL	NULL
30	#/publication#/book#/year	book	year	NULL	NULL
:	:	:	:	:	:

Fig. 5. Path table

Observation 1. For an element e_n, if $e_1/e_2/e_3/\ldots/e_{(n-1)}/e_n.table = null$, e_n is not mapped to a table. If $e_1/e_2/e_3/\ldots/e_{(n-1)}/e_n.table \neq null$ and $e_1/e_2/e_3/\ldots/e_{(n-1)}/e_n.column = null$, e_n is mapped to a table name.

Observation 2. The $e_1/e_2/e_3/\ldots/e_{(n-1)}/e_n.parentCode$ represents the parent table of $e_n.table$ on the path from e_1 to e_n if $e_1/e_2/e_3/\ldots/e_{(n-1)}/e_n.table \neq null$.

Observation 3. If $e_1/e_2/e_3/\ldots/e_{(n-1)}/e_n.recursions \neq null$, a recursion occurs on the path from e_n to $e_n.table$.

4 Experiments

We experimented to compare the performance of our approach with those of *Shared* and *Hybrid* approaches[10]. We used ten DTDs and queries of which the length of the

path is 3[15][16]. We compared the performance in terms of the number of joins and the number of subqueries per query.

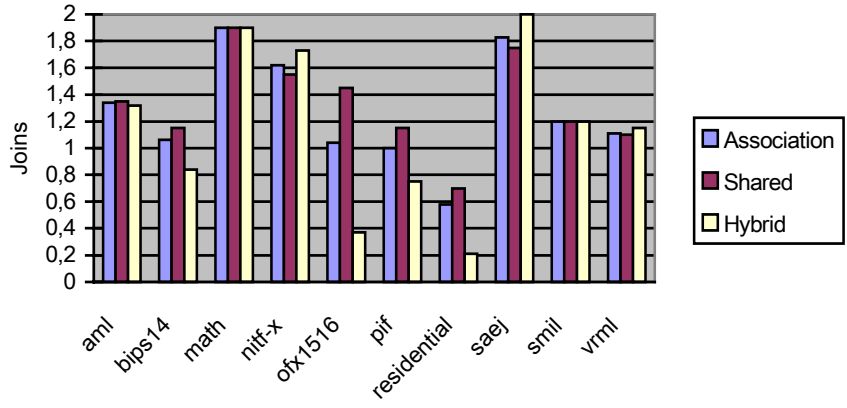


Fig. 6. Comparison of the number of joins

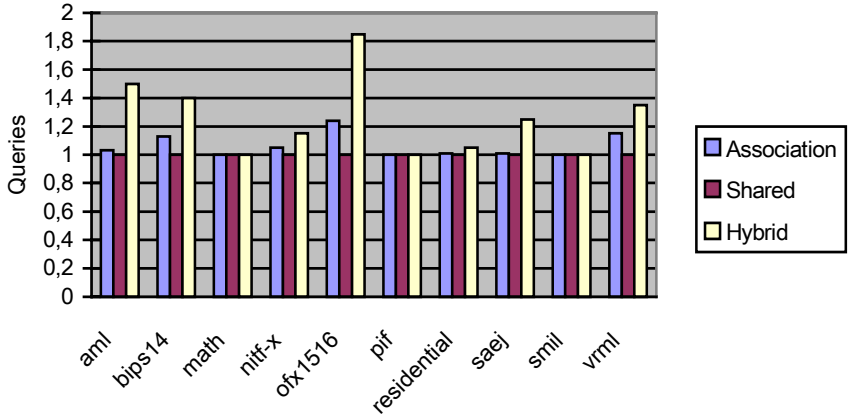


Fig. 7. Comparison of the number of subqueries per query

Association inlining has the number of joins less than 92.21% of that in *Shared inlining* as shown in Fig. 6. Fig. 7 shows that the number of subqueries in our approach is less than 88.05% of that in *Hybrid inlining*.

In our approach, the number of joins is reduced compared to *Shared inlining* and the number of subqueries per query is reduced compared to *Hybrid inlining*. The reduction rate depends on the characteristics of DTDs. When the number of nodes that are mapped to separate tables in *Shared inlining,* but inlined in our approach, is large, the number of joins is largely reduced. When the number of nodes that are inlined in *Hybrid inlining*, but mapped to separate tables in our approach, is large, the

number of subqueries per query is apparently reduced. However, some results of the experiments show that the performance is very similar to each other irrelevant to inlining methods. This is mainly due to the fact that three inlining methods have the same processing methods of sub-elements with set values and elements with recursions.

5 Conclusion

In this paper, we have proposed a new inlining method called the *Association inlining* for mapping DTDs to relational tables. It extends *Shared inlining* and *Hybrid inlining* to reduce relational fragments and excessive joins. We define various rules for mapping the nodes of a DTD graph to relational tables. *Association inlining* creates the smaller number of tables compared to *Shared* and *Hybrid inlining* by eliminating element-only elements that are not required to be mapped to tables. It has reduction in the number of joins and the number of subqueries per query to some extents compared to *Shared inlining* and *Hybrid inlining* respectively. The reduction is achieved by the discriminative processing of elements whose in-degree is greater than one.

Additionally, we store the schema information by using a *Path table*, which contains relational schema information of the path from the root to every element. It is exploited in processing queries over XML data. We are going to experiment with various datasets large enough and different queries to evaluate the performance of our method and to compare to *Shared* and *Hybrid inlining*.

References

1. Carey, D., Florescu, D., Ives, Z., Lu, Y., Shanmugasundaram, J., Shekita, E., Subramanion, S.: XPREANTO: Publishing Object-Relational Data as XML. Informal Proceedings of the International Workshop on the Web and Databases (2000) 105-110
2. David, M.M.: ANSI SQL Hierarchical Processing Can Fully Integrate Native XML. SIGMOD Record, Vol. 32, No. 1. (2003) 41-46
3. Fernandez, M., Kadiyska, Y., Morishima, A., Suciu, D., Tan, W.C.: SilkRoute: A Framework for Publishing Relational Data in XML. ACM Transactions on Database Systems (2002) 438-493
4. Florescu, D., Kossmann, D.: Storing and Querying XML Data Using an RDBMS. IEEE Data Engineering Bulletin, Vol. 22, No. 3. (1999) 27-34
5. Funderburk, J.E., Kiernan, G., Shanmugasundaram, J., Shekita, E., Wei, C.: XTABLES: Bridging Relational Technology and XML. IBM Systems Journal (2002) 616-641
6. Lu, S., Sun, Y., Atay, M., Fotouhi, F.: A New Inlining Algorithm for Mapping XML DTDs to Relational Schemas. The 1st International Workshop on XML Schema and Data Management held in conjunction with ER2003 (2003) 366-377
7. Shanmugasundaram, J., Kiernan, J., Shekita, E., Fan, C., Funderburk, J.: Querying XML Views of Relational Data. Proceedings of the 27th VLDB Conference (2001) 261-270

8. Shanmugasundaram, J., Shekita, E., Barr, R., Carey, M., Lindsay, B., Pirahesh, H., Reinwald, B.: Efficiently Publishing Relational Data as XML Documents. Proceedings of the 26th VLDB Conference (2000) 65-76
9. Shanmugasundaram, J., Shekita, E., Kiernan, J., Krishnamurthy, R., Viglas, E., Naughton, J., Tatarinov, I.: A General Technique for Querying XML Documents Using a Relational Database System. SIGMOD Record, Vol. 30, No. 3. (2001) 20-26
10. Shanmugasundaram, J., Tufte, K., He, G., Zhang, C., Dewitt, D., Naughton, J.: Relational Databases for Querying XML Documents: Limitations and Opportunities. Proceedings of the 25th VLDB Conference (1999) 302-314
11. Shin, B.J., Jin, M. Storing and Querying XML Documents Using a Path Table in Relational Databases. The 1st International Workshop on XML Schema and Data Management held in conjunction with ER2003 (2003) 285-296
12. Williams, K., Brundage, M., Dengler, P., Gabriel, J., Hoskinson, A., Kay, M., Maxwell, T., Ochoa, M., Papa, J., Vanmane, M.: Professional XML Databases. Wrox Press (2000)
13. Yoshikawa, M., Amagasa, T.: XRel: A Path-Based Approach to Storage and Retrieval of XML Documents Using Relational Databases. ACM Transactions on Internet Technology, Vol. 1, No. 1. (2001) 110-141
14. Zhang, X., Pielech, B., Rundensteiner, E.A.: XML Algebra Optimization. Technical Report WPI-CS-TR-02-25, Worcester Polytechnic Institute (2002)
15. W3C Recommendation. XML Path Language (XPath) Version 1.0. In http://www.w3c.org/TR/xpath (1999)
16. W3C Recommendation. XQuery 1.0: An XML Query Language. In http://www.w3c.org/TR/xquery/ (2003)

XCRAB: A Content and Annotation-Based Multimedia Indexing and Retrieval System*

SeungMin Rho[1], SooCheol Lee[1], EenJun Hwang[1], and YangKyoo Lee[2]

[1] Graduate School of Information and Communication
Ajou University, Suwon, Korea
[2] Department of Civil and Environmental Engineering
Daelim College, Anyang, Korea

Abstract. During recent years, a new framework, which aims to bring a unified and global approach in indexing, browsing and querying various digital multimedia data such as audio, video and image has been developed. This new system partitions each media stream into smaller units based on actual physical events. These physical events within each media stream can then be effectively indexed for retrieval. In this paper, we present a new approach that exploits audio, image and video features to segment and analyze the audio-visual data. Integration of audio and visual analysis can overcome the weakness of previous approach that was based on the image or video analysis only. We implement a web-based multimedia data retrieval system called XCRAB and report on its experiment result.

1 Introduction

With the advances in storage technology and the advent of the World Wide Web, there has been an explosion in the amount and complexity of digital information being generated, analyzed, stored, accessed and transmitted.

Most of the data is multimedia in nature, including digital images, video, audio and simple text data. To manage and handle this vast amount of multimedia, we need techniques to efficiently retrieve information over large multimedia repositories based on their content. Due to the difficulty in capturing the content of multimedia objects using textual annotations and the non-scalability of the approach to large data sets (due to high degree of manual effort required for the annotations), the approach based on content-based retrieval over visual features has become a promising research direction. This is evidenced by several prototypes and commercial systems that have been built recently.

Images can be treated in many ways. One approach[8] is to consider visual properties of images such as color, texture, shape, and so on. This allows us to ask queries such as “Find all images that are mainly red in color” or “Find all images which has a mix of colors similar to the example image.” The “visual features”-based

* This research was supported by University IT Research Center Project.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 859–868, 2004.
© Springer-Verlag Berlin Heidelberg 2004

approach has been actively pursued for some time and has yielded some useful systems such as QBIC[8].

Another approach[10, 11] is to consider the semantic composition of images in terms of the individual objects contained in them and the spatial relationships among them. This would enable us to ask queries such as "Find images that show a red apple on a table" or "Find images that show a river running below a rough stone bridge." Note that these queries incorporate visual properties as well as spatial relationships between objects.

We assume that the raw multimedia data is in the form of physical programs typically consisting of a combination of one or more of the following media types: visual, audio and text. In these media, events can be extracted at different levels of abstraction. Such events may or may not exist physically in the data stream.

In this paper, we propose solutions for a coherent system design that facilitates more sophisticated search and retrieval needs. Our proposed solutions cover system architecture, multimedia data storage and indexing, database and its remote access. We then design, develop, and implement a system for multimedia data retrieval that utilizes aforementioned principles. It is also hoped that the proposed system can offer a concrete example of capabilities required from the MPEG-7 system and serve as a useful tool in the MPEG-7 experimentation model and working draft development.

The organization of the rest of the paper is as follows. In Section 2 presents indexing framework in XCRAB. Section 3 illustrates the XCRAB system architecture and scene determination process. Section 4 presents the system implementation and demonstrates some experimental results. Finally, Section 5 concludes the paper.

2 Indexing Framework of XCRAB

One of the most important issues in the design of multimedia database system is the representation of image contents that allows for efficient retrieval of multimedia data through a user-friendly interface. In this section, we discuss different media type indexing techniques.

2.1 Audio Feature Analysis

The current audio processing researches [1, 2, 3, 4, 5] have produced numerous techniques for audio analysis. Most of these techniques model the human vocal system and extract relevant features such as pitch, energy, zero crossing rate, pause rate, and so on. Use of spectrogram is also popular in audio signal analysis. In this paper, we describe how to compute features for MPEG audio data in five different ways [6].

Short time average energy is one of the important features in audio processing. The average energy indicates the loudness of the audio signal. It is easy to separate the voice and noise signals using the average energy function. The short time energy function is defined by

$$E_m = \frac{1}{N}\sum_{n=0}^{N-1} x(n)^2 * h(m-n), \tag{1}$$

where m is the number of samples, $x(n)$ is the input signal and $h(m)$ is a processing window function that provides a linear filter with impulse response. To reduce the spectral leakage, we decide to use the hamming window function instead of the square window. In this case, $h(m)$ is defined by

$$h(m) = \begin{cases} 0.54 - 0.46\cos(\frac{2\pi m}{N-1}) & for\ 0 \le m \le N-1 \\ 0 & otherwise \end{cases} \tag{2}$$

The zero crossing rate (ZCR) indicates the frequency of signal amplitude sign change. To some extent, it indicates the average signal frequency. The average zero crossing rate is calculated as follows:

$$ZCR = \frac{\sum_{n=1}^{N} |\operatorname{sgn} x(n) - \operatorname{sgn} x(n-1)|}{2N} \tag{3}$$

where sgn $x(n)$ is the sign of $x(n)$ and is 1 if $x(n)$ is positive and -1 if $x(n)$ is negative. The average zero crossing rate can be used to distinguish between voiced and unvoiced speech signals, because unvoiced speech components normally have much higher ZCR values than voiced ones. If the zero crossing rate is high, the speech signal is unvoiced while if the zero crossing rate is low, the speech signal is voiced.

The signal information, which has significant high frequency components, is useful for audio classification because music normally has higher frequency components than speech. Therefore, it is important to calculate low and high frequency band energy. The ratio of low frequency energy and the high frequency energy is an important measure in voiced and unvoiced signal detection.

The bandwidth indicates the frequency range of a sound. Music normally has a higher bandwidth than speech signals. The simplest way of calculating bandwidth is by taking the frequency difference between the highest frequency and lowest frequency of the non-zero spectrum components. In some cases, “non-zero” is defined as at least 3 dB(decibel) above the silence level.

A harmonic sound consists of a series of major frequency components including the fundamental frequency, which is the lowest frequency. The harmonicity of a sound can be determined by checking the frequencies of dominant components if the frequencies are multiples of the fundamental frequency. Sounds from most musical instruments are harmonic. Most environmental sounds such as applause, footstep and explosion are nonharmonic.

2.2 Image Indexing Scheme

Image contents can be described in terms of either the image objects and their spatial relationships or the objects in the original scene and their spatial relationships[11]. In the first case, 2D objects are involved, and 2D spatial relationships are evaluated

directly on the image plane. In the second case, scenes associated with images involve objects that differ greatly in their structural properties from one application to the other. Specifically, scenes involve objects if objects have prevalently a 2D structure or involve 3D objects if they are common real-world scenes. Spatial relationships for the two cases are 2D and 3 D, respectively.

ROIs(Region of Interest) for querying image databases are special images defined through a computer system by the user. They maintain a symbolic relationship with objects in the real world. According to the type of application, either a 2D or 3D structure can be associated with each icon to build virtual scenes with 2D or 3D objects, respectively. Every image in the database contains a set of unique and characterizing image objects that scatter in any arbitrary locations. There could exist various spatial relationships among these image objects. The spatial location[7] can be represented either by the relative coordinate or by the absolute coordinate.

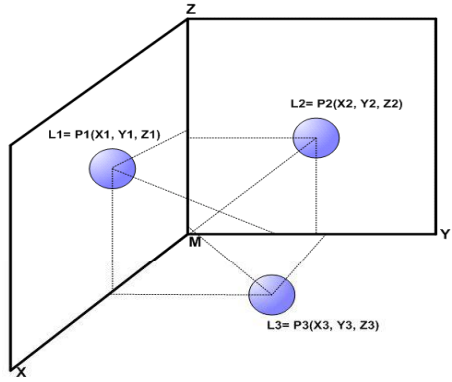


Fig. 1. Spatial location points

In a 3-D space, the spatial location of an object O in an image is represented as a point P_o where $P_o = (X_o, Y_o, Z_o)$, and an image itself as a set of points $P=\{P_1, P_2,\ldots, P_n\}$, where n is the number of objects of interest in the image. These points are tagged or annotated with labels to capture any necessary semantic information of the object. We call each of these individual points representing the spatial location of an image object a *spatial location point*[7]. For the sake of simplicity, we assume that the spatial location of an image object is represented by only single spatial location point, and hence, the entire image is represented by a set of spatial location points. In order to represent spatial relations among the spatial location points of an image, we decompose an image into four equal size quadrants. Fig. 1 shows an image whose three spatial location points are located at the different quadrants.

This representation scheme is translation, orientation and scale independent. Based on this scheme, we can define image location algebra for image objects X, Y and Z as shown in Table 1.

***Definition* 1**. A spatial graph is a set of pair (V, E) where:

- $V = \{L1, L2, L3,\ldots, Ln\}$ is a set of nodes, representing objects.
- $E = \{e1, e2, e3,\ldots, en\}$ is as set of edges, where each edge is connecting two nodes L1 and L2, and labeled with a spatial relationship between them.

In the figure, some of the spatial relationships among the spatial location points are represented along their edges using image location algebra. Hear, M is a special spatial point called fiducial point.

Table 1. Image location algebras

Notation	Operator	Meaning
X < Y	Lupper	X is located in Left upper of Y
X ∧ Y	Llower	X is located in Left lower of Y
X > Y	Rupper	X is located in Right upper of Y
X ∨ Y	Rlower	X is located in Right lower of Y
X ∪ Y	Upper	X is located in Upper of Y
X ∩Y	Below	X is located in Below of Y
X]Y	Right	X is located in Right of Y
X [Y	Left	X is located in Left of Y
X / Y	Center	X or Y is located in Center of M
X % Y	Overlap	X is Overlapped of Y
X ⊗ Y	Inside	X is Inside of Y
X ⊕ Y	Outside	X is Outside of Y
X • Y	In front of	X is In front of Y

According to the definition 1, the spatial relationships for the figure are:

$V = \{L1, L2, L3, L4\}$

$Rel = \{L1 \cup M\}, \{L1 > L2\}, \{L1 \wedge L3\}, \{L1 \bullet L4\}, \{L2 \wedge M\}, \{L2 \wedge L1\}, \{L2 \wedge L3\}, \{L2 \wedge L4\}, \{L3 \vee M\}, \{L3 < L1\}, \{L3 < L2\}, \{L3 \vee L4\}, \{L4 \bullet L1\}, \{L4 > L2\}, \{L4 < L3\}$

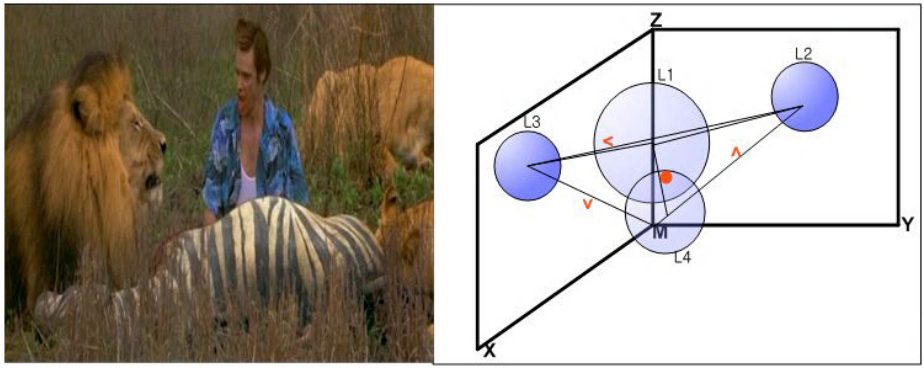


Fig. 2. Original image and its spatial graph

3 The XCRAB System

The architecture for our prototype XCRAB system is shown in Fig. 3. The XCRAB system consists of four major components: Shot Analyzer, Image Analyzer, Classifier, and Annotation Tool. Video data can be analyzed by identifying scenes/shots and extracting their key-frames. Traditional video analysis techniques are based on visual features such as color histogram or representative color in video frame. More recent studies have reported that audio-visual can improve the video analysis. Even though scene boundary detection can be done automatically with good accuracy, its semantic content interpretation still needs human intervention.

In this paper, we propose a scheme for automatic segmentation and annotation of audiovisual data based on both audio and video content analysis. In the video shot analysis, we first detect the video shot boundaries using the traditional shot detection methods. If current frame is detected as a video shot boundary, it is marked as the key frame. After scanning the whole video sequence, all key frames are extracted. Due to the coarse cut detection in a video sequence, some video shots may be very short. In order to overcome this problem, the short shot is merged with its preceding shot before the audio analysis if a shot consists of less than 10 frames.

In the image analysis, users define region-of-interest (ROI) directly on an image from video analyzer in order to make the spatial analyzer recognize clearly the intended content that could possibly represent only a subset or partial aspect of the images. Image retrieval is either sketch-based or content-based. A sketch-based query is in the form of drawing a sketch containing rough object outlines in some spatial arrangement and identifying each object. A content-based query is in the form of writing object information such as color and shape on the query interface. In both cases, the query is transformed into a spatial graph, which is then used as the basis for the matching process.

In the audio shot analysis, we divide the audio data into different shots based on the five different audio features, namely, energy function, average zero crossing rate, energy distribution, bandwidth and harmonicity. Each shot is classified as one of the following audio types: silence, pure speech, music, speech with music, environmental sound, and speech with environmental sound.

We first identify silent shots using the following criteria before the last classification processing. We measure both average energy threshold and zero crossing rate to detect silence. If the short time energy function is continuously lower than the average of certain set of thresholds or if an average zero crossing rate in the segment is lower than a certain threshold value, then the segment is indexed as "silence." For each frame, if its average energy threshold is below 3 and the zero crossing rate is above 50, we consider it is a silent frame. Within each shot, if the percentage of the silent frame is higher than 70%, this shot is considered as a silent shot, which will be ignored in the later processing.

In the last classification step, we determine the video scene boundaries by integrating the audio and video information. We characterize the video scenes as the semantic scenes using the information from the audio and video shots. After the characterization, audio/video shot information and semantic scenes are automatically annotated and stored in the database. Fig. 3 shows the overall procedure.

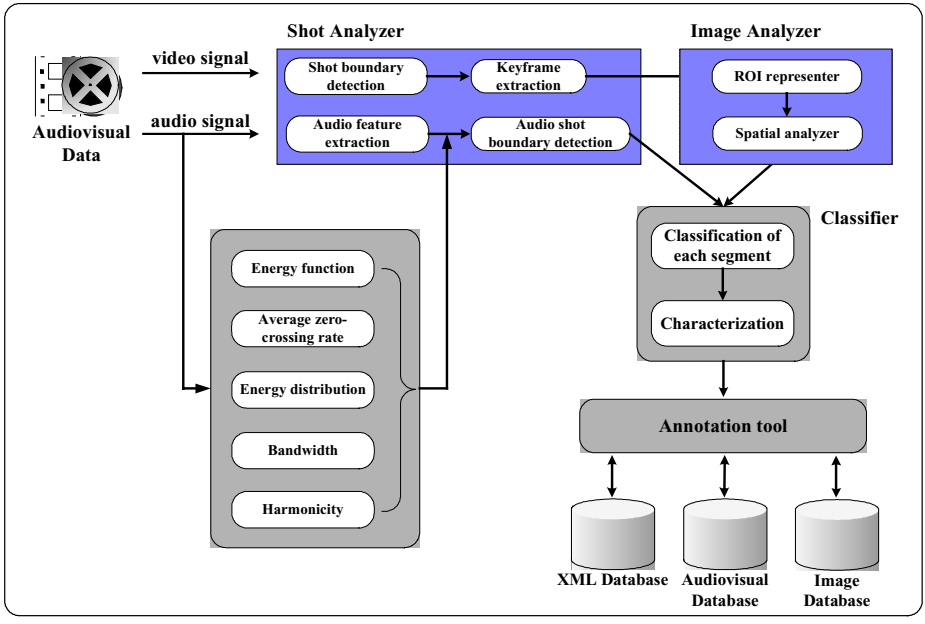


Fig. 3. XCRAB system architecture

4 Implementation

We have implemented a prototype multimedia retrieval system based on the indexing schemes that we described so far. It provides flexible user interface for query formulation and result browsing. Both client and server side are implemented using Java Applet and JSP. We used a set of JAI APIs, JMF, and jMusic for extracting image, video and audio features, and the eXcelon database system for handling meta-data in XML.

4.1 User Interface

The user interface is used to formulate queries and to generate scene representation either via development of an animation or through manual mark-up of objects on a real video, followed by image and audio indexing schemes.

Fig. 4 shows the snapshots of the XCRAB user interface. User may formulate queries using composite conditions such as keyword, spatial relationship, RGB value and audio genre and easily choose the colors from the palette as shown in Fig. 4-1. User also can easily select the type of audio, which have additional significant information, as shown in Fig. 4-2. Fig. 4-3 shows a spatial similarity and annotation-based query interface. For each image object marked by a rectangle, its edge is extracted and any necessary semantic information is coded into an XML document.

The query result represents contents of the document in the tabular form and the embedded video on the web browser as shown in Fig. 5. Video browsing is based on the hierarchical structure of video, so the user can play a specific video shot by selecting its representative frame(R-Frame). Fig. 5 shows the query result for a query of "Any" of the chosen audio sample and RGB values.

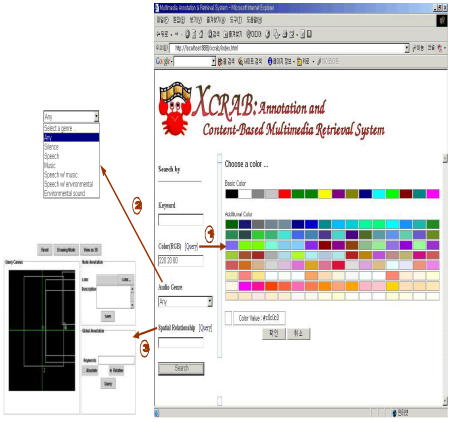


Fig. 4. User interface

Fig. 5. Query result

4.2 Experiments

The system is running on the Windows 2000 and the video data is stored in a RAID storage system to minimize its data loss and to improve data transfer rate. We have chosen a collection of video segments including movie, sports, news clips for the experiments. Using the XCRAB, we have tested some of the typical queries under different combinations of stored annotation, spatial constraint and audio information of video objects. Table 3 shows some of the results for a query would be:

Q1: "*Find the video shots with a red house in front of the tree*"

Q2: "*Find the video shots have a man and woman singing in the rain and wear a red hat*"

In the query Q1, retrieve all house shots in the database that were in front of the tree and have a red color. There are 5 shots in the collection that satisfy the query. In the table 2, queries 1-3 used keywords search only, query 4 used a combination of color, keywords and 2D spatial constraint and query 5 used combination of color, keywords and 3D spatial constraint. Using the keywords "house and tree" in conjunction with the spatial constraint "in front of" gives best precision.

In the table 3, query Q2 returns all man and woman shots in the database that were singing in the rain and wear a red hat. There were 24 images in the collection that satisfy the query. Experiments have revealed that retrieving shots based on keywords

Table 2. Query Q1 result

#	Spatial Constraint	Color	Keywords	Retrieved	Relevant	Precision
1	"-"		Tree	215	5	2.3
2	"-"	Red	House	47	5	10.6
3	"-"	Red	House and Tree	21	5	23.8
4	Left or Right	Red	House and Tree	13	5	38.5
5	In front of	Red	House and Tree	5	5	100

only gives marginal results. However, when incorporating spatial constraint and audio information in the query gives much better result.

Table 3. Query Q2 result

#	Audio	Spatial Constraint	Color	Keywords	Retrieved	Relevant	Precision
1	"-"	"-"	"-"	Man	786	24	3.0
2	"-"	"-"	"-"	Woman	358	24	6.0
3	"-"	"-"	Red	Man and Woman	142	24	17
4	"-"	"-"	Red	Man, Woman and Hat	85	24	28.2
5	"-"	Left or Right	Red	Man, Woman and Hat	57	24	42.1
6	"-"	Overlap or Left or Right	Red	Man, Woman and Hat	43	24	56
7	Singing	Left or Right	Red	Man, Woman and Hat	18	24	75
8	Singing	Overlap or Left or Right	Red	Man, Woman and Hat	24	24	100

5 Conclusion

In this paper, we described the use of annotation and segmentation for the indexing and retrieval of multimedia data. Our goal is to simplify the task of finding segments of video data suitable for inclusion in multimedia documents. To identify coherent portions of the media, each media was segmented, while annotations were performed for contents-based search.

We have built a prototype system called XCRAB based on the proposed indexing schemes and performed experiments to see how it works for typical multimedia retrieval requests. The XCRAB system provides tools for analyzing, annotating, querying, and browsing images in a user friendly way. Experimental results show the effectiveness of the overall indexing and retrieval framework. Our framework gives a significant query gains and provides a robust and generic solution for multimedia indexing, annotation and retrieval.

References

1. Hao Jiang, Tony Lin, Hongjiang Zhang, "Video segmentation with the Support of Audio Segmentation and classification," ICME'2000-IEEE International Conference on Multimedia and Expo, New York City, NY, USA, July 30 - August 2, 2000.
2. A. Yoshitaka, and M. Miyake, "Scene Detection by Audio-Visual Features," IEEE International Conference on Multimedia and Expo (ICME01), pp.49-52, 2001.
3. Dongge Li, Ishwar K. Sethi, Nevenka Dimitrova, Thomas McGee, "Classification of general audio data for content-based retrieval," Pattern Recognition Letters, vol. 22(5), pp. 533-544, 2001.
4. Shu-Ching Chen, Mei-Ling Shyu, Wenhui Liao, and Chengcui Zhang, "Scene Change Detection By Audio and Video Clues," IEEE International Conference on Multimedia and Expo (ICME02), pp.365-368, 2002.
5. S. Rho and E. Hwang, "FMF(Fast Melody Finder): A Web-based Music Retrieval System," Lecture Notes in Computer Science, Springer-Verlag, Vol.2771, pp.179-192, 2003.
6. S. Rho and E. Hwang, "Video Scene Determination using Audiovisual Data Analysis," Proc. of the 24th International Conference on Distributed Computing Systems (ICDCS'04) Workshops - Multimedia Network Systems and Applications (MNSA'04), Tokyo, Japan, March 2004, to appear.
7. S. Lee and E. Hwang, "Spatial Similarity and Annotation-Based Image Retrieval System", IEEE Fourth International Symposium on Multimedia Software Engineering, Newport Beach, CA, December 2002.
8. W. Niblack, et al. "The QBIC project: Query images by content using color, texture and shape," SPIE V 1908, 1993.
9. V. E. Ogle and M. Stonebraker, "Chabot: Retrieval from a Relational Database of Images," IEEE Computer, Vol. 28, No. 9, September 1995.
10. M. J. Egenhofer and R. D Franzasa, "Point set topological spatial relations," Journal of Geographical Information Systems, vol 5, no 2, 161-174, 1991.
11. S. Chang, Q. Shi and S. Yan, "Iconic indexing using 2-D strings," IEEE Trans. on Pattern Analysis & Machine Intelligence, Vol. 9, No. 3, pp. 413-428, 1987.

An Efficient Cache Conscious Multi-dimensional Index Structure

Jeong Min Shim[1], Seok Il Song[2], Young Soo Min[1], and Jae Soo Yoo[1]

[1] Department of Computer and Communication Engineering, Chungbuk National University
48 Gaesin-dong, Cheongju Chungbuk, Korea
{jmshim, minys, yjs}@netdb.chungbuk.ac.kr

[2] Department of Computer Engineering, Chungju National University, Iryu Meon Gumdan Lee, Chungju Chungbuk, Korea
sisong@chungju.ac.kr

Abstract. Recently, to relieve the performance degradation caused by the bottleneck between CPU and main memory, cache conscious multi-dimensional index structures have been proposed. The ultimate goal of them is to reduce the space for entries so as to widen index trees, and minimize the number of cache misses. They can be classified into two approaches according to their space reduction methods. One approach is to compress minimum bounding regions (MBRs) by quantizing coordinate values to the fixed number of bits. The other approach is to store only the sides of MBRs that are different from their parents. In this paper, we investigate the existing multi-dimensional index structures for main memory database systems through experiments under the various work loads. Then, we propose a new index structure that exploits the properties of the both techniques. We implement existing multi-dimensional index structures and the proposed index structure, and perform various experiments to show that our approach outperforms others.

1 Introduction

As the cost of main memory in server computer systems becomes cheaper, main memory DBMSs (MMDBMSs) prevail over various application areas. It is a well-known fact that MMDBMSs provide order-of-magnitude performance gain over traditional disk based DBMSs. The major bottleneck of traditional disk DBMSs was disk I/O, and many researches proceeded to hide the disk I/O that occupies most of fraction of total execution time of transactions. Consequently, the impacts of disk I/O on the performance of DBMSs are significantly reduced.

In the similar fashion, recently, as the performance gap between CPU and main memory gets larger, it becomes increasingly important to consider cache behavior and to reduce L2 cache line misses for the performance improvement of MMDBMSs [1], [2]. Subsequently, several researches to improve the performance of index structures for MMDBMSs by reducing L2 cache misses have been done actively in the database community [3], [4], [5], [6], [7], [8]. Since the end of 1990s, cache conscious index

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 869–876, 2004.
© Springer-Verlag Berlin Heidelberg 2004

structures have been one of the primary concerns to improve the performance of MMDBMSs. Particularly, in the beginning of the 2000s, some cache conscious multi-dimensional index structures were proposed to enhance the performance of modern applications such as Geographical Information System (GIS)s and Location Based System (LBS)s based on MMDBMSs.

To our knowledge, most recent cache conscious multi-dimensional index structures are cache conscious R-tree (CR-tree)[7], partial R-tree(PR-tree)[8] and normal R-tree[9] with a small node size, a cache line size or its multiple. CR-trees compress MBRs by quantizing coordinate values to the fixed number of bits. PR-trees store only the sides of MBRs that are different from their parents. The ultimate goal of both index structures is to reduce the space for MBRs so as to widen index trees and minimize the number of cache misses.

In this paper, we investigate existing multi-dimensional index structures for MMDBMSs such as CR-trees, PR-trees and normal R-trees through some experiments. Then, we propose a new index structure that exploits the properties of CR-tree and PR-tree. Actually, the partial MBR method of PR-tree works well when the number of entries is small, generally 3~7. However, the number of entries of CR-tree that uses compression techniques is much more than that of normal R-tree. In that reason, partial MBR method seems to be inadequate for CR-tree. Through experiments, we investigate that integrating CR-tree with PR-tree is valuable. Finally, we perform extensive experiments to show that our proposed index structure outperforms CR-tree, PR-tree and R-tree in various environments.

This paper is organized as follows. In section 2, we describe existing cache conscious uni-dimensional and multi-dimensional index structures. In section 3, we analyze PR-tree and CR-tree in detail, and propose PCR-tree. Section 4 presents experimental results that proposed algorithm outperforms existing index structures through various experiments. Finally, section 5 concludes this paper.

2 Related Works

Cache-conscious R-tree (CR-tree) and partial R-tree (PR-tree) are multi-dimensional index structures for MMDBMSs. We describe the features of the two index structures that form the basis of our proposed index structure.

In the PR-tree, usually the node size of R-trees in MMDBMSs is a cache line size (32~64 bytes) or the multiple of a cache line size. The authors of PR-trees performed an experiment to show how many MBRs in R-tree share their sides with their parents. According to the results of the experiment, on average about 1.75 sides of a MBR are shared with parent MBR.

In the CR-tree, the ultimate goal of partial key techniques is to widen index trees by eliminating redundant information from entries in a node. CR-trees also pursue the same goal as PR-tree but its approach is quite different. The basic idea of CR-tree is to widen index tree by compressing MBRs so as to make R-trees cache conscious.

3 Proposed Partial CR-Tree (PCR-Tree)

3.1 Analysis of CR-Trees and PR-Trees

PR-trees work well when the number of entries of a node is small enough (about 3~7). As the number of entries in a node increases, the ratio of the number of shared sides to total number of sides in a node is reduced. For instance, when the maximum number of entries in a node is 3 in 2 dimensional data spaces, the minimum number of sides that are shared with their parent MBR is 4. If the data type of each side is 4 bytes, 48 bytes are required for 3 MBRs. However, if we do not store the shared sides, only 32 bytes are required. In this example, 33% space is saved.

PR-trees need additional information to indicate whether a side is stored or not. For a 2 dimensional MBR, 4 bits are used, so additional 12 bits are needed for 3 MBRs. Finally, total space for 3 MBRs is 34.5 bytes. However, when the number of maximum entries of a node is 40, the minimum number of shared sides is 4. Subsequently, the total space for them is (640-16) bytes + (4*40) bits = 644 bytes. That is, no space gain is produced.

The MBR compression technique of CR-tree increases the fanout of nodes. When the cache line size is 64 bytes, a pointer to a child node group is 4 bytes, the space for the number of entries is 4 bytes and the flag to indicate whether the node is leaf or non leaf is 1 bytes, 39 bytes are used as the space for quantized relative representation of MBR (QRMBR)s. Therefore when the coordinate values are quantized into 4 bits, the maximum number of entries of a node is about 19. As we mentioned in the previous paragraphs, applying the partial MBR technique to CR-tree would be meaningless since the number of entries becomes larger than normal R-trees. However, the compression technique of CR-tree enlarges MBRs to align MBR to quantized coordinate values. Consequently, the probability of sharing sides of MBRs in a node with the parent MBR of the node may increase.

We performed experiments to count how many discriminators in a CR-tree share sides with their parents when the number of bits used for compression is 4 and 8 bits. The results in Table 1 are for a CR-tree with 19 and 9 entries per node when the number of bits used for compression is 4 and 8 bits, respectively.

Table 1. Ratios of shared sides in CR-trees

	4 bits compression	8 bits compression
Uniform	43.75 %	28.125%
Real	50 %	34.375%

When 4 bits are used for compression, on average about 45% of space for a MBR is saved. When 8 bits are used, on average about 30% of space for a MBR is saved. Additional information (bit fields) is not considered in the results. Considering the additional information, the average saved space is 20% for 4 bits compression and 18% for 8 bits compression. From these facts, we conclude that the combination of the partial technique and the compression technique would be meaningful.

3.2 Layout, Typeface, Font Sizes, and Numbering

The main idea of our proposed PCR-tree is the combination of the partial technique of PR-tree and the compression technique of CR-tree. Generally, it is well known that as MBRs become large, overall search performance is degraded in R-trees. However, CR-trees show that even though MBRs become large, fat index trees compensate the disadvantage and provide good search performance. Our approach is to widen CR-trees by applying the partial technique of PR-trees without any loss of accuracy of MBRs. The quantization levels are made the same for all nodes. The structure of our PCR-trees is shown in Fig. 1. NE denotes the current number of entries in a node. CP denotes the child node group for non-leaf nodes and records for leaf nodes. AMBR is the absolute MBR of a node. The MBRs of the node is recalculated relatively to this absolute MBR. BF denote bit fields of entries and ENTRIES denote entries. Each entry of ENTRIES includes a pointer to the record object.

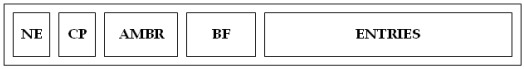


Fig. 1. Node structure of PCR-tree

NE which denotes the number of entries in a node plays another important role. On every insertion and deletion in a node, inserters or deleters check if the partial technique is meaningful for the node, i.e., total size of the node with partial technique is greater than that of the node without partial technique. If the partial technique is meaningless, we convert the NE as negative values. Traversers must note NE to read entries in an node with the partial technique correctly since to access ENTRIES they need to know the size of BF which is determined by the NE. However, if the partial technique is not applied to the node, BF does not have any value. Therefore, actually, BF is not needed and the negative value of NE indicates that this node does not have BF.

Insert operations proceed in two phases like other R-tree families. In the first phase, a leaf node where a new entry is placed is located. Once the leaf node is located, an inserter checks if the node is overflowed. If overflow occurs, the node is split. Otherwise, the inserter put the entry into the leaf node, and decides the node type of the leaf node, i.e., node type is *pnode* or *qnode*. *pnode* means that the node can partial MBRs and *qnode* means that we cannot apply partial technique to the node.

Overall search algorithm of the proposed method is the combination of CR-tree and PR-tree. However, when visiting a node, searchers must first check if the node type is *pnode* or *qnode*. If the node is *qnode*, i.e., the NE of the node is positive value, the search algorithm of CR-tree is applied. Otherwise, the both search algorithms of CR-tree and PR-tree are applied.

Once searchers visit a node, they first see the sign of NE in the node. The NE that denotes the number of entries in a node indicates that the node is *pnode* or *qnode*. The positive value of NE indicates that the node is *pnode*, so searchers must restore

all sides of MBRs of ENTRIES in the node by referring to BF and AMBR of the node. After that, they perform the search procedure of CR-tree.

4 Performance Evaluation

4.1 Description of Evaluation

To evaluate the performance of our proposed PCR-tree, we compared it with CR-tree, PR-tree and R-tree. We configured the node size of R-tree to multiple times of a cache line size. Our experiments were performed on a Pentium 4 1GHz with 256 Mbytes main memory under Windows 2000. The size of a level 2 cache line is 64 bytes. We implemented CR-tree, PR-tree, R-tree and our PCR-tree. Commonly, we used the quadratic splitting algorithm for all trees. Also, the bulk loading technique was implemented. The used technique is sort tile recursive (STR)[10].

Two kinds of data set such as uniformly distributed synthetic data set and real data set (TIGER) were used in our experiments. We obtained the TIGER data set from website[11] and generated the synthetic data set as follows. First we generated integers between 0 and 100 by using random number generator and stored them to a 2-dimensional array sequentially. Then we recursively constructed MBRs by grouping 2 elements of the array form the beginning. We did not limit the side length of a MBR so the areas of MBRs were varied.

We build trees with the STR bulk loading algorithm. Then, we measured search performance under various conditions. Generally, multi-dimensional index trees show quite different performance according as which tree construction method is used such as bulk loading and one-by-one insertion. Therefore, we measured search performance after inserting a certain number of entries one by one into bulk loaded trees. The number of entries inserted by one-by-one method was varied. Also, to measure insertion performance, we inserted 10,000 entries to bulk loaded index trees.

We generated 10,000 range queries and measured the number of node accesses, execution time and the number of cache misses with varying the size of range queries form 0.001 to 0.1. In our experimental results, we found that the search performance of the real data set is very similar to that of the synthetic data set. To save the space, we describe only the search performance results of the real data sets.

4.2 Experimental Results

Fig. 2 to 4 show the performance of search operations when the experiments are performed with real data set. In Fig. 2, the number of node accesses decreases as the node size increases. When the node size is 1024 bytes, most index structures access the similar number of nodes. Also, in Fig. 4, as the node size increases, the execution time increases. When the node size is 64 bytes, the performance of our PCR-tree(4) is best.

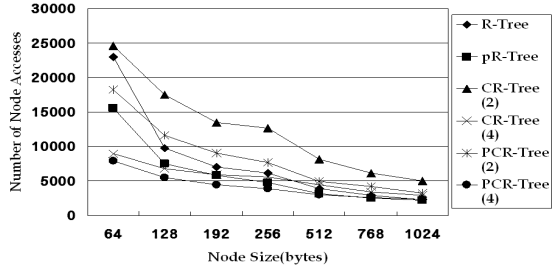


Fig. 2. Node accesses (real data, search)

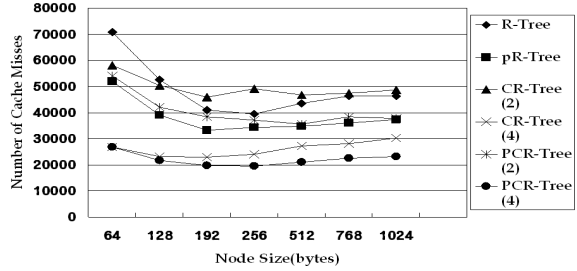


Fig. 3. Cache misses (real data, search)

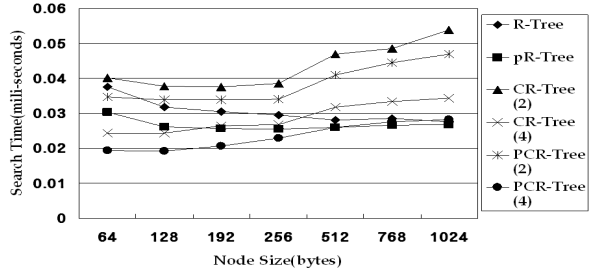


Fig. 4. Execution time (real data, range search)

Also, we measured the execution time of insert operations with inserting 10,000 entries in one-by-one fashion to bulk loaded index structures. Our PCR-tree have rather complex insert algorithm than those of the other index structures. PCR-tree need to determine whether the compression technique is meaningful and calculate relative coordinates. As shown in Fig. 5, insert time of R-tree is fastest in all cases. However, when the node size is 64 or128 bytes, the insert time of R-tree and PCR-tree is almost same and the search performance is best in 64 bytes nodes. Therefore, the insert performance of PCR-tree is the same to that of the others.

Fig. 6 shows the search performance of index structures after inserting a number of entries into bulk loaded index structures. As we mentioned in the beginning of Section 4, the performance of search operations of multi-dimensional index structures are different according to the insertion methods. This Figure is to show that PCR-tree works well even in one-by-one insertion. As shown in Fig. 6, our PCR-tree(4) outper-

forms others in all cases. The number of node accesses of PCR-tree(4) increases moderately, while others increase steeply.

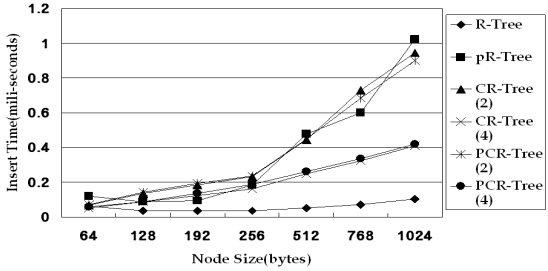


Fig. 5. Execution time (uniform distribution, insert)

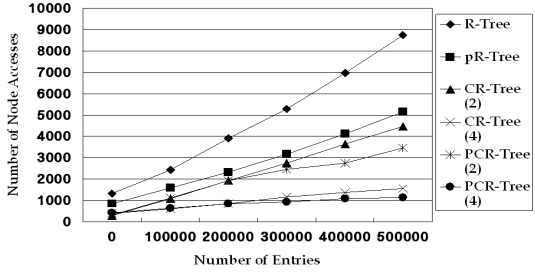


Fig. 6. Node accesses (uniform distribution, search, after dynamic insertion)

5 Conclusion

In this paper, we proposed a cache conscious multi-dimensional index structure that exploits the properties of existing methods. Through extensive performance comparisons under the various conditions, we show that the proposed PCR-tree outperforms the existing methods. Our contributions are summarized as follows. First, we investigated existing cache conscious multi-dimensional index structures. Second, we implemented existing multi-dimensional index structures and the proposed index structure, and performed various experiments to show that our approach outperforms others.

Acknowledgement. This work was partially supported by the Korea Research Foundation Grant(KRF-2003-041-D00489) and KISTEP.

References

1. Anastassia Ailamaki, David J. DeWitt, Mark D Hill and David A. Wood: DBMSs on a Modern Processor: Where Does Time Go?, In Proceedings of VLDB Conference (1999) 266-277
2. Stefan Manegold, Peter A. Boncz and Martin L. Kersten: Optimizing database architecture for the new bottleneck: memory access, In VLDB Journal 9(3) (2000) 231-246
3. Jun Rao and Kenneth A. Ross: Cache Conscious Indexing for Decision-Suppoert in Main Memory, In Proceedings of VLDB Conference (1999) 78-79
4. Jun Rao and Kenneth A. Ross: Making B+-trees Cache Conscious in Main Memory, In Proceedings of ACM SIGMOD Conference (2000), 475-486
5. Philip Bohannon, Peter Mcllroy and Rajeev Rastogi: Main-Memory Index Structures with Fixed-Size Partial Keys, In Proceedings of ACM SIGMOD Conference (2001) 163-174
6. Shimin Chen, Phillip B. Gibbons and Todd C. Mowry: Improving Index Performance through Prefetching, In Proceedings of ACM SIGMOD Conference (2001) 235-246
7. Kihong Kim, Sang K. Cha and Keunjoo Kwon: Optimizing Multidimensional Index trees for Main Memory Access, In Proceeding of ACM SIGMOD Conference (2001) 139-150
8. I. Sitzmann and P.J. Stuckey: Compacting discriminator information for spatial trees, In Proceedings of the Thirteenth Australasian Database Conference (2002) 167-176
9. Guttman, A.: R-trees: a Dynamic Index Structure for Spatial Searching, In Proceedings of ACM SIGMOD Conference (1984) 47-47
10. Scott T. Leutenegger, J. M. Edgington, Mario A. Lopez: STR: A Simple and Efficient Algorithm for R-tree Packing, In Proceedings of ICDE Conference (1997) 497-506
11. http://www.cs.du.edu/~leut/MultiDimData.html

Tracking of Moving Objects Using Morphological Segmentation, Statistical Moments, and Radon Transform

Muhammad Bilal Ahmad[1], Min Hyuk Chang[2], Seung Jin Park[3], Jong An Park[2], and Tae Sun Choi[1]

[1] Signal and Image Processing Lab, Dept. of Mechatronics,
Kwangju Institute of Science and Technology, Gwangju, Korea
{bilal, tschoi}@kjist.ac.kr
[2] Dept. of Information & Communications Engineering,
Chosun University, Gwangju, Korea.
japark@chosun.ac.kr
[3] Dept. of Biomedical Engineering,
Chonnam National University Hospital, Gwangju, Korea.

Abstract. This paper describes real time object tracking of 3D objects in 2D image sequences. The moving objects are segmented by the method of differential image followed by the process of morphological dilation. The moving objects are recognized and tracked using statistical moments. The straight lines in the moving objects are found with the help of Radon transform. The direction of the moving object is calculated from the orientation of the straight lines in the direction of the principal axes of the moving objects. The direction of the moving object and the displacement of the object in the image sequence are used to calculate the velocity of the moving objects. The simulation results of the proposed method are promising on the test images.

1 Introduction

Segmentation and tracking semantic objects in video are essential tasks for the most content-based digital video applications. Object tracking is an important problem in the field of content-based video processing. When a physical object appears in several consecutive frames, it is necessary to identify its appearances in different frames for purposes of processing. Object tracking attempts to locate, in successive frames, all objects that appear in the current frame. The most straightforward approach to this task is to consider objects as rectangular blocks and use traditional block matching algorithms [1]. However, since objects may have irregular shapes and deformations in different frames, video spatial segmentation and object temporal tracking can be combined [2]-[3].

In object tracking, pattern recognition is to deal with the geophysical data based on the information contained in the image sequences. An automatic interpretation or recognition of geophysical data is very difficult from the image sequences [4]. A lot of efforts have been found in the literature [5]-[9], and still a lot of research is needed for automatic recognition of moving objects in the image sequences. Most methods of object tracking such as optical flow [10], block matching [3], etc are highly

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 877–886, 2004.
© Springer-Verlag Berlin Heidelberg 2004

computational and hence difficult to apply in the run time applications. For object tracking, we are mostly interested in the rigid body motion. Every moving object, say, vehicles, walking human being, etc are all consists of rigid body motion. We can exploit this rigid body motion for tracking the objects.

In this paper, we have proposed an effective moving object tracking based on the orientation of the moving objects. Moving objects locations are found in the image sequence by the method of differential edge image followed by morphological dilation. After locating the moving objects in the image sequences, we extract different high-level features directly from the regions of pixels in the images, and describe them by various statistical measures. Such measures are usually represented by a single value. Measurements of area, length, perimeter, elongation, compactness, moments of inertia are usually called statistical geometrical descriptions [11]. We use the statistical geometrical descriptions to recognize the moving objects in the image sequences. The principal axes of inertia for the moving objects in the image sequences are used for extracting the direction of the moving objects. The straight lines in the moving objects are determined by the Radon transform [12]. The straight lines in the moving objects that are almost aligned with the principal axes are averaged to find the direction of the moving objects. We assumed that the velocity of the moving objects is not too high, and we restrict the search area for tracking of the individual moving objects within the most probable range.

This paper is organized as follows. Section 2 describes the segmentation of the moving objects using differential edge images followed by the process of morphological dilation. Section 3 describes the different statistical descriptors that will be utilized for tracking and recognizing the objects. Section 4 explains the Radon transform to find the direction of the moving objects. Simulation results are shown in section 5. At the end, we will conclude our paper with few final remarks.

2 Segmentation of Moving Objects

We first segment the moving objects in the input image sequence. Edge detector is applied on the two input image sequence. For removing the background (still part) in the images, we find the binary difference image from the resulting two input edge maps, as:

$$D(x, y) = ABS(E2(x,y) - E1(x,y)), \tag{1}$$

where E2(x,y), E1(x,y) are the two binary edge maps of the input image sequence, and D(x,y) is the resulting binary difference image. The resulting binary difference image D(x,y) gives us the possible location of moving objects. To find the areas of moving objects, we binary dilate the difference image D(x,y) as:

$$DL = dilate(D), \tag{2}$$

where DL is the dilated image of the difference binary image D. The dilated image DL detects the areas of moving objects in the image sequence. In the dilated image DL, all possible moving objects (both real and erroneous moving objects) are detected. The erroneous moving objects are detected due to the presence of noise in

the images. We applied a thresholding method to extract the real moving objects from the dilated image DL. We first label the moving objects in the dilated image DL and then calculate the binary areas of each of the moving objects. We threshold the real moving objects that have considerable area in the dilated image as:

$$\begin{aligned} &\text{if } A[DL(j)] > T_{area} \\ &\qquad \text{Real Moving Object (keep it)} \\ &\text{else} \\ &\qquad \text{Erroneous Moving Object (discard it)} \end{aligned} \tag{3}$$

where A[DL(j)] calculates the binary area of j^{th} labeled object in DL, and T_{area} is the threshold, the value of which depends on the size of input images, and the distance of camera from the scene. We discard the erroneous moving objects by replacing 1s with 0s in that area. Finally, we get the image, which contains only real moving objects in the image sequence. We then calculate the statistical descriptors in those actual moving areas.

3 Object Tracking Using Statistical Descriptors

After segmenting the moving objects from the input image sequence, a matching algorithm is needed between the regions in the two consecutive images for tracking and recognizing the moving objects. A region matching or similarity is obtained by comparing the statistical descriptors of the two regions. Since the images may have translational, rotational, and scaling differences (objects may move further or closer to the camera), the region or shape measures should be invariant with respect to translational, rotation and scaling. One kind of such invariants belongs to statistical moments, called statistical invariant descriptors.

3.1 Statistical Moments and Invariant Descriptors

The moment invariants are moment-based descriptors of planar shapes, which are invariant under general translation, rotational and scaling transformations. Such statistical moments work directly with regions of pixels in the image using statistical measures. Such measures are usually represented by a single value. These can be calculated as a simple by-product of the segmentation procedures. Such statistical descriptors usually find area, length, perimeter, elongation, Moments of Inertia, etc. The moments of a binary image b(x, y) are calculated as:

$$\mu_{pq} = \sum_{x} \sum_{y} b(x, y) x^{p} y^{q}, \tag{4}$$

where p and q define the order of moment. Where b(x,y) can be omitted as it has only 1 and 0 values, so sums are only taken where b(x,y) has values 1. The center of gravity of the object can be found from moments as:

$$\bar{x} = \frac{\mu_{10}}{\mu_{00}}, \qquad \bar{y} = \frac{\mu_{01}}{\mu_{00}}, \tag{5}$$

where $(\bar{x}, \bar{y})$ are the coordinates of the center of gravity. The pqth discrete central moment m_{pq} of a region is defined by

$$m_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q \tag{6}$$

where the sums are taken over all points (x,y). Hu [13] proposed seven new moments from the central moments that are invariant to changes of position, scale and orientation of the object represented by the region using central moments of lower orders. All the seven moments are translational, rotational and scale invariants. These invariants will help us in the object tracking of the moving objects.

The principal axes of inertia define a natural coordinate system for a region. Let θ be the angle that the x-axis of the natural coordinate system (the principal axes) makes with the x-axis of the reference coordinate system. Then θ is given by

$$\theta = \frac{1}{2} \tan^{-1}\left[\frac{2m_{11}}{m_{20} - m_{02}}\right] \tag{7}$$

From the principal axes of inertia, we can find the direction of the moving objects.

3.2 Tracking of Moving Objects

For tracking of moving objects, the seven statistical descriptors are calculated for the detected moving regions of the input image sequence. There are translation and rotation of moving objects due to motion from one image frame to another image frame, and also the object can move far or closer from the camera, which results in the different size of the object in terms of pixels for the fixed camera position. The next step is the comparison of the statistical descriptors in the two images. Here we have assumed that either the motion of the objects are very small, or the frame rate is very high, so that we can restrict the search area for tracking of the individual moving objects within the most probable range. With the help of the statistical descriptors, we recognize and track different kinds of moving objects. We found the statistical invariant descriptors for every detected moving region in the two images, and then track the moving objects within the search region by comparing the statistical descriptors.

4 Finding Velocity Vectors for the Moving Objects Using Radon Transform

After tracking the moving objects in the input image sequence, we determine the principle axes using Eq. (7) for each of the segmented moving objects. The principal

axes do not give the true direction of the moving object, because of 2D image representation of 3D objects. However, the principal axes give the rough estimate of the direction of the moving objects. To find the true direction, we need to determine the straight lines in the object. The Radon transform is used to find the straight lines in the moving objects.

4.1 Straight Lines Using the Radon Transform

Radon transform can be efficiently used to search the straight lines in the images. It transforms two dimensional images with lines into a domain of possible line parameters, where each line in the image will give a peak positioned at the corresponding line parameters. The Radon transformation shows the relationship between the 2-D object and the projections. Let us consider a coordinate system shown in Fig. 1. The function $g(s,\theta)$ is a projection of f(x,y) on the axis s of θ direction. The function $g(s,\theta)$ is obtained by the integration along the line whose normal vector is in θ direction. The value $g(0,\theta)$ is defined that it is obtained by the integration along the line passing the origin of (x,y)-coordinate. The general Radon transformation is given as:

$$g(s,\theta) = \int \int_{-\infty}^{\infty} f(x,y)\delta(x\cos\theta + y\sin\theta - \rho)dxdy \tag{8}$$

The Eq. (8) is called Radon transformation from the 2-D distribution f(x,y) to the projection $g(s,\theta)$.

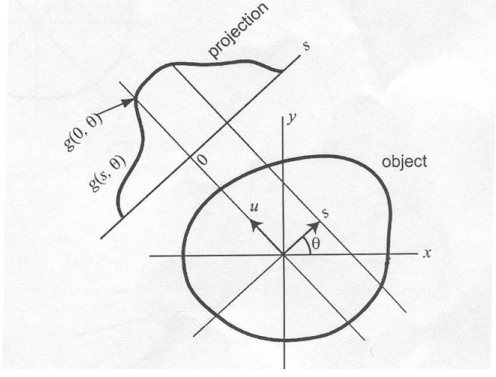


Fig. 1. Radon Transformation

Although the Radon transformation expresses the projection by the 2-D integral on the x,y-coordinate, the projection is more naturally expressed by an integral of one variable since it is a line integral. Since, the s,u-coordinate along the direction of projection is obtained by rotating the x,y-coordinate by θ, the Radon transform, after a change of axes transformation, is given as:

$$g(s,\theta) = \int \int_{-\infty}^{\infty} f(s\cos\theta - u\sin\theta, s\sin\theta + u\cos\theta)\delta(0)dsdu \quad (9)$$

Since the δ-function in Eq. (9) is a function of variable s, we get

$$\int_{-\infty}^{\infty} \delta(0)ds = 1$$

It follows from the above that the Radon transformation $g(s,\theta)$ in Eq. (8) is translated into the following integral of one variable u,

$$g(s,\theta) = \int \int_{-\infty}^{\infty} f(s\cos\theta - u\sin\theta, s\sin\theta + u\cos\theta)du \quad (10)$$

This equation expresses the sum of f(x,y) along the line whose distance from the origin is s and whose normal vector is in θ direction. This sum, $g(s,\theta)$, is called ray-sum.

The Radon transform could be computed for any angle, and could be displayed as an image. From Fig. 1, the Radon transform of the input images can be computed for any angle. In practice, we compute the Radon transform at angles from 0 to 179 degree, in 1 degree increment. The procedure to find the straight lines using the radon transform is as follows:

- Compute the binary edge image of input image using the edge detector
- Compute the Radon transform of the edge image at angles from 0 to 179
- Find the locations of strong peaks in the Radon transform matrix. The location of these peaks corresponds to the location of straight lines in the original image.
- The straight lines are drawn in the image space from the information obtained through the strong peaks in the Radon transform.

4.2 Object Orientation

We determined all the straight lines using the Radon transform for the every tracked object in the image sequence. The orientation of the moving object is determined from the straight lines and the principal axes of the object as shown in Fig.2. The x-axis of the principal axes is selected as the reference axis. The straight lines that make a greater angle than the threshold angle are discarded. The angles that the remaining straight lines in the object make with the principal axes are averaged. The average angle thus determined is the true orientation of the 3D moving objects.

The direction of moving object is found from the law of cosines from the orientation angles of the individual moving object in the two consecutive images. From Fig.3, we can find the direction of the moving object. In Fig.3, let L_1 and L_2 be the two lines making angle θ_1 and θ_2 with respect to x-axis of the reference frame, respectively. L_1 and L_2 correspond to the true orientation of the moving object as determined in Fig. 2.

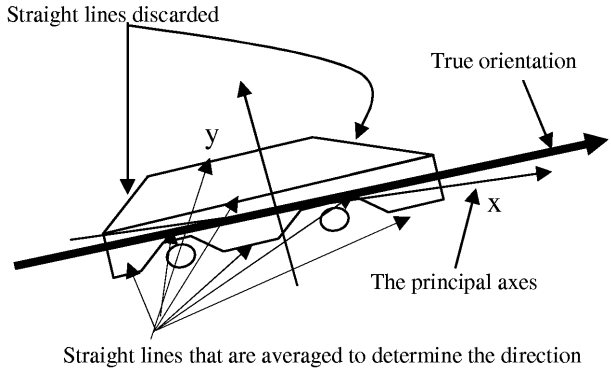


Fig. 2. Determining the true orientation of the object

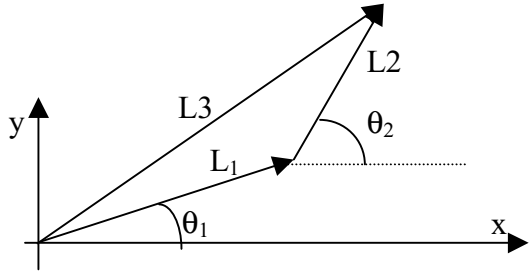


Fig. 3. Determining the direction of the moving object

The mathematical derivation for the moving object direction θ_3 with respect to x can be derived as:

For L1 : $y = m_1 x + c_1$

For L2 : $y = m_2 x + c_2$

By solving the above equations, the intersection point of L_1 and L_2 can be found as:

$$x_{\text{int}} = \frac{c_2 - c_1}{m_1 - m_2}, \qquad y_{\text{int}} = m_1 x_{\text{int}} + c_1 \tag{11}$$

The origin in Fig.3 is the center of gravity of the object in the previous image frame. From law of cosines

$$l_3^2 = l_1^2 + l_2^2 - 2 l_1 l_2 \cos(\pi + \theta_1 - \theta_2) \quad \text{and} \tag{12}$$

$$\cos(\theta_3 - \theta_1) = \frac{l_1^2 + l_3^2 - l_2^2}{2 l_1^2 l_3^2} \tag{13}$$

The angle θ_3 gives the direction of the moving object. The small l_1, l_2, l_3, are the magnitudes of L_1, L_2 and L_3 lines. For calculating the magnitude of the velocity vector, the Euclidean distance of the two centers of gravity is measured. From the

angle θ_3, and Euclidean distance of the centers of gravity, we calculate the velocity vectors of the moving objects. Same method is applied for extracting the velocity vectors of each individual moving object.

5 Simulation Results

For simulation, 256 x 256 gray-level image sequences are used. One test sequence is shown in Fig.4. First we segment the moving objects from the input image sequence using the proposed differential edge algorithm. The mask obtained in the Fig. 5(a) is used to segment the moving objects from the input image sequence as shown in Fig. 5(b). The statistical descriptors are calculated for the segmented moving regions only. The moving objects are recognized using the similarity of statistical descriptors. The direction of the moving object is determined using the Radon transform and the principal axes. The principal axes as shown in Fig. 5(b) doesn't give the right

Fig. 4. A test sequence

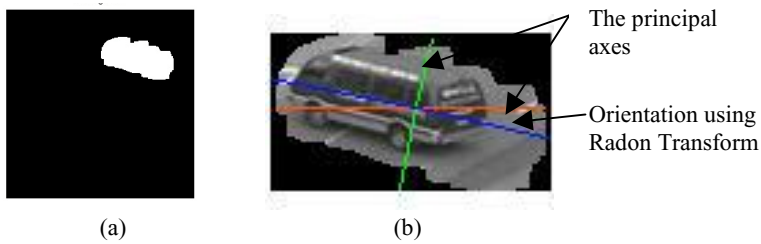


Fig. 5. (a) The mask for segmenting the moving objects (b) the direction and the principal axes of the object.

Fig. 6. Object tracking using the proposed algorithm on three test image sequences

direction of the 3D object, whereas the direction obtained by using the Radon transform represents more accurate direction of the moving object. Figure (6) shows the tracking result of three test image sequences. The three test moving objects are accurately tracked in the image sequences.

6 Conclusions

In this paper, a new algorithm is proposed for segmenting, recognizing, tracking and finding the velocity vectors for moving objects in a video stream. There are many popular techniques for finding velocity vectors, such as optical flow, and block matching algorithm, but they are time-consuming algorithms. Our method is computationally fast and gives compact information about the moving objects. From the input video stream, we segment the moving objects using the edge differential algorithm. For tracking of the moving objects, we proposed method based on the statistical invariant moments or descriptors, which are invariant to translation, rotation and scaling transformation. After tracking, we found the orientation of the moving objects using the principal axes of inertia and the Radon transform. From the knowledge of the orientation of the moving object in the consecutive image frames, we found the direction of the moving objects. From the displacement of the center of gravity, we found the Euclidean distance of the moving objects. The final velocity vector for a moving object is calculated from the orientation angles, and the Euclidean distance of the centers of gravity of the object. The process of edge detection and segmentation accurately find the location and areas of the real moving objects, and hence the extractions of moving information are very easy and accurate. The orientation of the objects is more accurately determined from the Radon transform.

Acknowledgements. This work was supported by the Korea Research Foundation Grant (KRF-2003-041-D20470).

References

1. A.M. Tekalp, Digital Video Processing, Parentice Hall, 1995.
2. R.C. Gonzalez and R. E. Woods, Digital Image Processing, Prentice Hall, 1993.
3. Berthold Klaus Paul Horn, Robot Vision, McGraw-Hill, 1986.
4. N. Diehl, “Object Oriented Motion Estimation and Segmentation in Image Sequences,” Signal Processing: Image Communication, Vol. 3, No. 1, pp. 23-56, Feb. 1991.
5. C. Cafforio and F. Rocca, "Tracking Moving Objects in Television Images," Signal Processing, Vol. 1, pp. 133-140, 1979.
6. Willium B. Thompson, “Combining motion and contrast for segmentation,” IEEE Trans. Pattern Anal. Machine Intelligence, pp. 543-549, Nov. 1980.
7. M. Etoh et. al., "Segmentation and 2D motion estimate by region fragments," Proc. 4th Int. Conf. Computer Vision, pp.192~199, 1993.

8. P.J. Butt, J.R. Bergen, R. Hingorani, R. Kolczinski, W.A. Lee, A. Leung, J. Lubin, and H. Shvaytser, “Object tracking with a moving camera, an application of dynamic motion analysis,”. in IEEE Workshop on Visual Motion, pp. 2-12, Irvine, CA, March 1989.
9. Chao He, Yuan F. Zheng, and Stanley C. Ahalt, “Object tracking using the Gabor wavelet transform and the golden section algorithm,” IEEE transactions on multimedia, vol. 4, No. 4, December 2002.
10. B. K. P. Horn and B. G. Schunck, "Determining optical flow," Artificial Intelligence,. 17, pp.185~203, 1981.
11. Robert M. Haralick, Linda G. Shapiro, Computer and Robot Vision, vol. 1, Addison Wesely, 1992.
12. S. R. Deans, The Radon Transform and some of its applications, Kreiger, 1983.
13. M. K. Hu, “Visual pattern recognition by moment invariants,” IEEE Trans. Information Theory, Vol. IT-8, No. 2, pp. 179-187, 1962.

Feature Extraction and Correlation for Time-to-Impact Segmentation Using Log-Polar Images*

Fernando Pardo, Jose A. Boluda, and Esther De Ves

Dpt. Informática - Universidad de Valencia Avda. Vte. Andrés Estellés s/n, 46100, Burjassot Spain
(Fernando.Pardo, Jose.A.Boluda, Esther.Deves)@uv.es
http://tapec.uv.es/

Abstract. In this article we present a technique that allows high-speed movement analysis using the accurate displacement measurement given by the feature extraction and correlation method. Specially, we demonstrate that it is possible to use the time to impact computation for object segmentation. This segmentation allows the detection of objects at different distances.
There are several methods to measure movement in front of a mobile vehicle (robot) equipped with a camera. Some methods detect movement from the analysis of the optical flow, while other methods detect movement from the displacement of objects or part of the objects (corners, edges, etc). Those methods based on the optical flow are suitable for high speed analysis (say 25 images per second) but they are not very accurate and treat the image as a whole, being it difficult to separate different objects in the scene. Those methods based on image feature extraction are good for object recognition and clustering, that can be more precise than other methods, but they usually require many calculations to yield a result, making it difficult to implement these methods in a navigation system of a robot or mobile vehicle.

1 Introduction

Time to impact is the time that a moving camera will take to impact an object (or vice versa). This time to impact can be directly calculated from the optical flow (object speed on the focal plane) [1]. Calculating the time to impact for each object in a scene, allows a moving platform to detect objects and their distances, making it possible a 3D reconstruction of a scene.

The calculation of the time to impact becomes simpler if log-polar images [2] are employed [3]. Fig. 1 shows the transformation of the scaling of an approaching object into a linear displacement. The focal plane (camera) and the computational plane (array in the computer memory) are shown in this figure. The original object (black ring) is a centered ring at the focal plane, but it is converted

* This research has been funded by the Spanish Ministerio de Ciencia y Tecnología project TIC2001-3546 and EU FEDER

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 887–895, 2004.
© Springer-Verlag Berlin Heidelberg 2004

to a straight line in the computational plane after the log-polar transformation. The scaling produced by the camera approaching the ring is converted to just a displacement in one of the orthogonal axis (radial coordinate). This interesting property can be exploited to simplify computations of such approaching movements, commonly found in the movement of robots toward an objective (time to impact).

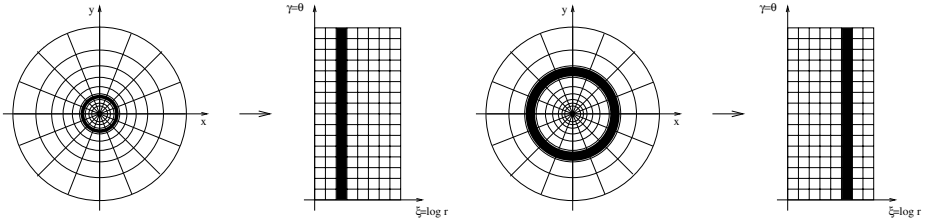


Fig. 1. Camera approaching to an object (black ring) in log-polar coordinates

In log-polar imaging, the time to impact computation simplifies to just a division between the radial spatial gradient and the temporal gradient. A first approach for time to impact computation is to calculate these spatial and temporal derivatives to obtain a time to impact value for each point in the image. Some experiments have shown that this approach gives noisy and not precise results. There are several reasons for this: the spatial gradient must be constant for a certain image region, this gradient must be high for grey level error minimization, also temporal gradient must be high for the same reason, temporal gradient depends on the object speed, etc. It is common to obtain very different time to impact values even for the same object. It is necessary to apply some statistical analysis to obtain the time to impact in a region over the time, and even this analysis does not yield precise results. Nevertheless, this differential method has shown its utility when fast calculating global time to impact of an image.

Other possibility, for the time to impact calculation, is to detect motion by one dimension feature correlation [4]. One refinement of this method consists of tracking the movement of some features of the image over the time. This last approach has several advantages: no matter whether the speed of the object is low or high, it is simple to detect erroneous objects, some feature extraction mechanisms can be very fast and finally it is accurate. The disadvantage is that it usually takes a relatively large amount of time to calculate results and keep track of objects. Fortunately one of the advantages of log-polar vision is the selective reduction of information, since it concentrates pixels in the image center (most interesting part) and decreases resolution toward the periphery (least interesting part).

A resolution of 76 rings by 128 pixels per ring has been used for the experiments [5]. This makes a total of 9.728 pixels to be processed. Comparing this

data (roughly 10 K), to the data of a standard 512x512 Cartesian image (256 K) the difference is quite significant. The difference in the number of pixels to be processed has a direct impact on the rates at which the images can be processed. This time saving can reach several orders of magnitude while the precision is still kept at acceptable values depending on the performed image analysis.

2 Time to Impact Computation

The time to impact of an approaching object is the time required by an object to reach the camera. If the camera moves, this time to impact is the time employed by the camera to reach the first object in its trajectory. This time to impact can be calculated from the measurement of object speeds at the sensor plane. The use of log-polar coordinates simplifies this calculation as it is shown next.

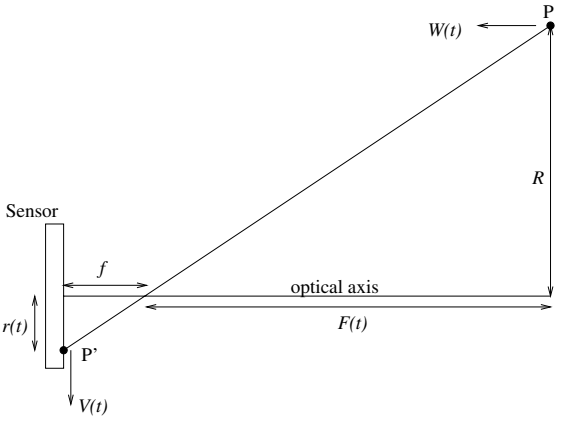


Fig. 2. Pin-hole model of an object approaching the camera

Fig. 2 shows the approaching object (P) projection on the camera sensor plane (P') following the pin-hole camera model. From this figure the following relation is straight forward:

$$\frac{f}{r(t)} = \frac{F(t)}{R} \tag{1}$$

where $F(t)$ is the object (P) projection distance to the camera focus, R is the distance of the object to the camera optical axis, f is the camera focus distance and $r(t)$ is the distance of the object projection (P') to the optical axis. $F(t)$ and $r(t)$ depend on the time since the object (or the camera) is moving. The speed at which the object P is approaching the camera is $W(t) = \mathrm{d}F(t)/\mathrm{d}t$, and the resulting speed of P' in the sensor plane is $V(t) = \mathrm{d}r(t)/\mathrm{d}t$. It is possible to obtain a relation between these two speeds deriving equation (1) with respect to time:

$$-\frac{f}{r^2(t)}V(t) = \frac{W(t)}{R} \tag{2}$$

In the other hand, the time to impact τ, supposing a constant approaching speed $W(t)$, can be evaluated by:

$$\tau = -\frac{F(t)}{W(t)} \tag{3}$$

Now it is interesting to have an expression, for the time to impact, that only takes into account camera or image parameters. For this objective we can take equation (1) to obtain an expression for $F(t)$ and equation (2) to have an expression for $W(t)$ and then substitute in equation (3) to obtain:

$$\tau = \frac{r(t)}{V(t)} \tag{4}$$

This is to say that the time to impact computation of an approaching object can be calculated as the division of the radius of the object projection (image) and the object projection speed. Both measurements can be obtained directly from image analysis.

Both magnitudes $r(t)$ and $V(t)$ are vectors that may have any spatial orientation depending on the approaching object motion. It means that, supposing a Cartesian representation, it is necessary to take into account the velocity components in X and Y directions.

Now we can consider a camera with a log-polar sensor which pixel distribution follows this equation:

$$r(t) = Ae^{B\xi(t)} \tag{5}$$

where $\xi(t)$ is the radial component of the log-polar computational plane as shown in Fig. 1. This expression can be derived to obtain the velocity in the log-polar domain:

$$V_\xi(t) = \frac{1}{B}\frac{V(t)}{r(t)} \tag{6}$$

The equation for the time to impact computation in log-polar coordinates is obtained substituting this expression in equation (4):

$$\tau = \frac{1}{B}\frac{1}{V_\xi(t)} \tag{7}$$

where B is the constant exponential growth factor of the log-polar transformation and $V_\xi(t)$ is the object projection speed in the radial direction measured in the log-polar computational plane. There are two advantages of the log-polar transformation compared to the Cartesian representation [6]: first, there is no need of knowing the object position in the image, since $r(t)$ is cancelled and it does not appear in the equation; and second, only one component (radial) of the velocity must be calculated, since we are supposing movement along the optical axis. This last feature is especially important since it simplifies the amount of calculus to be performed.

Equation (7) shows that it is enough to calculate $V_\xi(t)$ (radial optical flow) to obtain the time to impact in log-polar coordinates. This radial velocity $V_\xi(t)$ can be calculated as $\mathrm{d}\xi/\mathrm{d}t$ and can be approximated by $\Delta\xi/\Delta t$. The $\Delta\xi$ value is calculated tracking the movement of a point from image to image, and Δt is the time between two images (if the time unit is given in images then $\Delta t = 1$). The following sections explain the calculation of $\Delta\xi$ and the necessary filtering to obtain sound time to impact results.

3 Feature Extraction

Some methods for feature extraction have been tried and the fastest and simplest method yielded the best results for this particular application. Explanation of these feature extraction tried methods follows.

There are few feature extraction methods that can be employed in the log-polar domain, due to the special mathematic characteristics of this mapping. One of these problems is the distortion suffered by objects after transformation (the shape of any object is not constant and depends on its position in the log-polar plane). Nevertheless, there is a characteristic of the log-polar mapping that allows the utilization of some object detection methods: since the log-polar is a conformal transform, angles in Cartesian coordinates are preserved in the log-polar coordinate system. It is then possible to employ a feature extraction method based on corner or junction detection. This allows the measurement of relevant point movement and object detection despite the fact that the object changes as it moves (or the camera moves).

One of these methods consists of a 2D grey-level detector based on a statistical analysis of the gradient orientations in a circular neighbourhood of the point considered as a possible 2D feature [7]. This method is not computationally expensive and is more robust, especially for the log-polar mapping, than other corner detectors.

According to this approach, a corner point p can be defined as a point in the image whose gradient is not null and for which the orientations of the edges that converge in it are grouped around two (or more) different modes. Thus, it is proposed the hypothesis that a corner, where n edges converge, can be modelled as a mixture of n von Mises distributions. Therefore, the method to detect 2D features is to study the distribution of the orientations and to test the null hypothesis: the hypothesis that the distribution of orientations constitutes a mixture of two von Mises distributions. If the hypothesis can not be rejected, the point is assumed to be a corner where two edges converge. The test used for this purpose is the Watson-Stephens test.

The time to calculate this feature extraction is not too high; nevertheless, the number of significant points in the image is not high enough to perform a statistical analysis to obtain accurate results. This statistical analysis is still necessary since this feature tracking is not enough to obtain accurate data (it has at least the error of the pixel size).

In order to obtain more representative points, a more local feature has been considered. Something simple, as the maximum spatial gradient of a predefined

area, has been tried. This feature takes less computation time and gives more representative points than the other. This approach is good for accurate results, but it is still possible to simplify the feature extraction even further.

Taking into account that approaching movements parallel to the optical axis, thus linear in the log-polar domain, generate radial optical flow in the focal plane, it is possible to use a different feature for tracking. It has been made the assumption that any approaching object would move along a radial axis in the focal plane. Therefore, it is just enough to keep track of the object edge along the radial coordinate; even more, it is not necessary that this edge is the corresponding edge of a real object: anything traversing some grey threshold could be considered an edge. So, this edge has been calculated binarizing the image and taking the spatial derivative in the radial direction of the binarized image. The disadvantage of this method is that it is only good for time to impact computation and radial movements.

This last feature extraction has shown similar or even better results than the others with less calculation, so all following results have been obtained using the radial edge feature extraction.

4 Feature Tracking

This feature detector procedure (edge detector in this case) is followed by a matching process, which looks for correspondences between current and past interest points. Assuming that interest points have been located in all images of a sequence, the correspondence between points in consecutive images can be found by using the assumption of maximum velocity. This assumption implies that a point in the image will correspond to the closest point in the next image. Using this simple idea we have constructed an algorithm to track edge points in the scene: for each point of a frame we find the closest point in the next image.

It is possible that new interest points appear and others disappear at every new frame, due to image noisy, image change, etc. A new interest point appears when no track close enough is found in previous images. A point track could be cancelled when there is not a near point in next frame. When a new point appears, a new track is created for it; when a track disappears, it is considered not present in the image anymore. Calculated data on a given time is taken from points that were tracked at least for a given number of frames before.

Fig. 3 shows four selected images from the sequence employed for the experiments. This sequence has a total of 281 images and corresponds to the camera approaching toward two objects located at difference distances. The closest object is around 145 images far and the furthest is 290 images away; it takes almost the whole sequence to impact this last object. The log-polar computational plane is shown along the real Cartesian image.

The tracking of some points could be wrong due to mismatches obtained by the proposed algorithm and image noise. However, some points are tracked correctly and their movements behave as expected. All wrong tracked points are removed in a successive step; in fact, more than 700 interest points appear in the sequence of 281 images. After wrong point elimination, only tens remain.

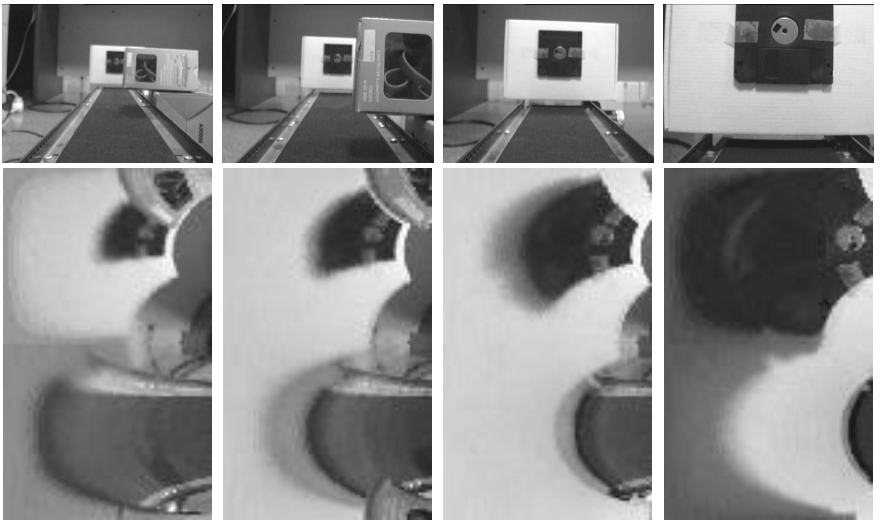


Fig. 3. Images 30, 100, 170 and 240 of total sequence (181). Original (up) and the corresponding log-polar (down)

5 Time to Impact Results and Object Segmentation

The theory involving time to impact computation is very simple, but the real implementation of such a theory is not so easy [8]. The discrete nature of the images and the acquisition itself produce errors with similar magnitude as the parameters to be measured. A large change in the image (say a displacement of several pixels) is necessary to obtain accurate speed measurement and thus accurate time to impact. But such a displacement means that the object is moving too fast or the image rate is too low. In both cases there is no use of calculating the time to impact since it could crash immediately after measurement. Time to impact should be measured accurately with some time in advance. It is necessary the use of statistical analysis over a large amount of images to obtain an accurate time to impact calculation long before the approaching object become dangerous.

One of the reasons for choosing the edge as a feature for tracking is that it generates lots of interest points to track. It gives many points, so it is possible to calculate the time to impact over a large amount of points, obtaining a more precise result. The problem is that many of these tracked points are wrong due to the image discrimination and lack of point matching among frames of the sequence.

The first step for calculating accurate time to impact consists of discarding wrong tracked points. There is a first filtering where points that have been tracked for less than a fixed number of frames are discarded. The higher the frame number the sounder the track, but it cannot be very high to have a representative number of points. A value of 20 frames has been fixed for this

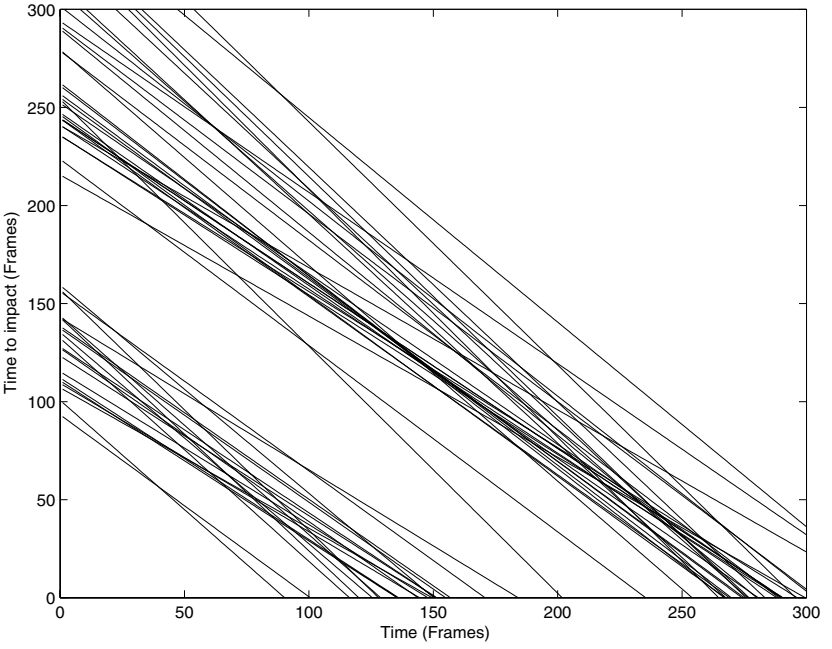


Fig. 4. Time to impact fitting for sound tracked points of the sequence.

experiment. Even cancelling all these tracks, there are still almost one thousand tracked points.

The real discrimination is performed making this important assumption: supposing object or camera speed constant, the time to impact of an object must decrease at every frame in one unit (supposing time is measured in frames). It is a basic assumption but it is always true for constant speed approaching movements: if an object has a time to impact of say 20 frames, in the next frame, the time to impact will be 19, and after 20 frames it must impact (time to impact equals cero). So, if the time to impact of an object is calculated at every frame, this time to impact must decrease linearly at a rate of one frame per frame. This property gives a straight decreasing line of 45 degrees (slope=-1) when representing the time to impact of an object versus time (see Fig. 4). Any object that is far from this behaviour can be discarded.

The assumption of constant approaching speed is basic for this algorithm. So it must be taken into account when using this algorithm in a mobile platform that could change the speed at anytime.

In the practical case of the present experiment, all points with slopes in the range of [-1.3:-0.7] have been accepted while all others have been rejected. This interval of accepted curves has shown to be appropriated for obtaining enough and accurate tracks. Doing this, we have reduced the total tracked points to just 51 that are more than enough for making a statistical refinement.

The last step consists of detecting the real objects at which the tracked points belong. The least squares fitting of these 51 points are shown in Fig. 4. It is possible to see that all lines are distributed around two places. These two areas correspond to the two objects of the experiments. Making segmentation from here is straight forward, just calculating the cutting point of the lines with the horizontal axis gives the time to impact of every object from the beginning. These times to impact forms a distribution with two clear heaps in this case; each one corresponds to each object of the experiment. The mean values of the two heaps have been obtained using the K-means algorithm. Only two or four iterations, depending on the initial points taken, are necessary to obtain the mean values. These values are 141 and 285 frames. The real times to impact of the two objects are 145 and 290, so the algorithm may still be improved, but it is good enough to segment both objects and give a result that is close to the real one.

6 Conclusions

An algorithm to detect objects at several distances from a sequence of images taken from a mobile platform has been presented. This algorithm successfully employs the time to impact computation to segment objects according to their distances to an approaching camera. We have shown that simple edges from binary images are enough as a feature extraction for point tracking. It has been also shown that feature tracking is a good enough method for time to impact computation, giving better results than previously experimented differential methods.

References

1. Daniilidis, K.: Optical flow computation in the log-polar plane. In: Int. Conf. on Computer analysis of images and patterns, CAIP'95. (1995) 65–72
2. Rojer, A., Schwartz, E.: Design considerations for a space variant visual sensor with complex logarithmic geometry. In: Proc. Int. Conf. on Pattern Recognition, Philadelphia, PA (1990)
3. Questa, P., Sandini, G.: Time to contact computation with a space-variant retina-like CMOS sensor. In: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'96, Osaka, Japan (1996)
4. Ancona, N., Poggio, T.: Optical flow from 1d correlation: Aplication to a simple time-to-crash detector. International Journal of Computer Vision **14** (1995) 131–146
5. Pardo, F., Dierickx, B., Scheffer, D.: Space-variant non-orthogonal structure CMOS image sensor design. IEEE Journal of Solid State Circuits **33–6** (1998) 842–849
6. Tistarelli, M., Sandini, G.: On the advantages of polar and log-polar mapping for direct estimation of time-to-impact from optical flow. IEEE Trans. on PAMI **15** (1993) 401–410
7. Díaz, M., Domingo, J., Ayala, G.: A grey-level 2d feature detector using circular statistics. Pattern Recognition Letters **18** (1997) 1083–1087
8. Pardo, F., Boluda, J.A., Coma, I., Mico, F.: High-speed log-polar time to crash calculation for mobile vehicles. Image Processing & Communications **8–2** (2002) 23–32

Object Mark Segmentation Algorithm Using Dynamic Programming for Poor Quality Images in Automated Inspection Process

Dong-Joong Kang[1], Jong-Eun Ha[2], and In-Mo Ahn[3]

[1] Mechatronics Engineering, Tongmyong University of Information Technology, 535, Yongdang-dong, Nam-gu, Busan, Korea
djkang@tit.ac.kr
[2]Multimedia Engineering, Tongmyong University of Information Technology, 535, Yongdang-dong, Nam-gu, Busan, Korea
jeha@tit.ac.kr
[3]Division of Computer and Electrical Engineering, Masan College, Masan, Kyungnam, Korea,
imahn@masan.ac.kr

Abstract. This paper presents a method to segment object ID (identification) marks on poor quality images under uncontrolled lighting conditions of automated inspection process. The method is based on multiple templates and normalized gray-level correlation (NGC) method. We propose a multiple template method, called as ATM (Active Template Model) which uses a search technique of multiple templates from model templates to match and segment character regions of the inspection images. Conventional Snakes algorithm provides a good methodology to model the functional of ATM. To increase the computation speed to segment the ID mark regions, we introduce the Dynamic Programming based algorithm. Experimental results using real images from automated factory are presented.

1 Introduction

As the manufacturing industry is being advanced, it has been increased the need of the automation for conventional inspection that are manually performed. In order to provide the visual inspection system applicable to the usual automation environment, it is necessary to develop the algorithm stable and performing reliable pattern recognition. The image segmentation as a main step to split meaningful areas from an interesting image is regarded as the most necessary and important procedure in the machine vision system. Many studies about the image segmentation have been being done so far and several results have been reported in previous works [2,3,8-10].

Conventional methods for image segmentation could be categorized to three parts such as the area based, the edge based, and the histogram based techniques [8-10]. In the area-based method, pixels with similar intensity values are considered as one area and thus consist of homogenous regions form an image. The split-and-merge algorithm stands for the area based segmentation [9]. The algorithm divides an image into

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 896–905, 2004.
© Springer-Verlag Berlin Heidelberg 2004

fixed areas and compares intensity similarity with neighboring pixels. If those areas are thought as the same area, merge those areas and if not, splits those areas iteratively. This method could provide good results but needs high time-complexity as well. The edge-based segmentation can segment the image into contour and the inner component by using intensity discontinuity of images. In the case of any noises found, the method may follow the false edge and hence needs any additional processing.

Segmentation by histogram [3] is mainly used when the distribution of the gray level is simple. The method makes the image segmentation easy by carrying out quantization the whole image as the highest two gray level after obtaining the histogram of all gray level. Thus this method may have a good segmentation effect when the distribution of histogram is concentrated on the two gray levels that present background and object, respectively. Especially, in case of object ID inspection of the automation process, the algorithms using histogram projection and Hough transformation are general approaches [5-6]. Most of such existing algorithms may show a good result in clean image with good visualization whereas the segmentation in poor quality images may be a serious problem. Such difficulty is caused by vague distribution characteristics of image intensity, due to noise and irregular lighting condition.

This paper aims to automatically segment the character area for recognition of object ID marks under poor visualization condition. The segmentation of inspection area is pre-processing procedure for recognition and the successful execution of this task will determine the success of whole process. Under poor lighting condition, none of the image features may be detected with reliability. Thus conventional approaches such as histogram projection and image binarization become difficult to be applied.
We first try to solve the ID mark segmentation by using template matching method because many conventional and commercial machine vision systems try to solve the mark recognition from application of template matching technique. However, the method is also difficult to apply in case of poor visualization image for the reason of frequent occurrence of fault matches.

We observed usual object ID marks are not single character and a few characters are continuously written on object surface to be inspected. If we try to match only single character by conventional template matching technique, it may be fail in most cases. However, assume that one model template consists of two characters, then it is possible to get more increased success rate of the matching. This idea is starting point for the work in this paper. Because the distance between any two neighboring characters of ID marks is not uniform and changeable, the functional for the variable distance has to be incorporated to a model equation. We have introduced the Snake's algorithm to object ID segmentation problem. The Snakes algorithm is called Active Contour Model because the spline curve of the Snakes actively responses to image data and attaches to strong image edges denoting object boundaries [7].

The functional to be optimized includes the distance between two neighboring characters and similarity ratio matched to each model character templates. The ATM (Active Template Model) proposed in this paper actively interacts to image data to segment successively positioned several characters in poor quality images. Dynamic programming is applied to reduce computational cost in optimization of the defined

functional. The experimental results using real images of factory automation (FA) process are presented to show the feasibility of the proposed algorithm.

2 Active Template Model

A central preprocessing step of conventional OCR algorithm is to segment character regions from background of inspection images. However, it not easy to apply the preprocessing step in poor quality images of FA processes. For an example, Fig. 1 is an image we try to tackle, the intensity value of the characters are not darker and brighter than the background. Hence, the usual approach in character recognition that consists of simple thresholding followed by extraction of connected components will not work [6]. Also noises and intensity variations from dynamic environments in factory make difficulties to solve the character recognition problem in the target images. As the most images in FA environment including examples of Fig. 1 present poor quality visualization, conventional methods such as region segmentation, boundary tracing, or binarization could not adequate to be applied for the segmentation of the ID marks under uncontrolled and dynamic lighting condition.

Fig. 1. Example of poor quality images

2.1 Snakes Algorithm

We introduce Snakes algorithm to solve the ID character segmentation problem in bad visualization images given from FA processes. The snake algorithm is based on the concept of active contours, which can be described as energy minimizing splines [7]. The concept of active contour is used very broadly in several previous papers and the related works [1,4,7]. It is an actively responding curve to the image features.

Kass [7] has proposed a model called Snakes (i.e., active contour models) as an active spline reacting with image features. The contour is initially placed near an edge under consideration, and then image forces draw the contour to the edge in the image. As the algorithm iterates, the energy terms can be adjusted to obtain a local minimum. That is, basic snake model is a controlled continuity spline under the influence of image forces. The internal spline forces serve to impose a piecewise smoothness constraint. The image forces push the snake toward salient image features. See the interaction between image data and active contour in Fig. 2. The active contour is represented by a vector, $v(s) = (x(s), y(s))$ having arc length s as parameter. Energy functional for the contour is defined by:

$$E_{\text{snake}} = \int_0^1 E_{\text{internal}}(v(s)) + E_{\text{external}}(v(s))ds \tag{1}$$

where E_{internal} represents the internal energy of the contour due to bending or discontinuities, E_{external} is the image forces. The image forces can be due to various events, e.g., lines, edges, and terminations. The internal spline energy is written:

$$E_{\text{internal}} = \alpha(s)\left|v_s(s)\right|^2 + \beta(s)\left|v_{ss}(s)\right|^2 . \quad (2)$$

The first-order continuity term has larger values where there is a gap in the curve, and the second-order curvature term will be large where the curve is bending rapidly.

The value of α and β at a point determines the extent to which the contour is allowed to stretch or bend at that point. If α is 0 at a point, a discontinuity can occur, while if β is 0, a corner can develop. The minimum energy contour is determined using techniques of variational calculus [7] or a neighbor region search of the snake control points [1]. Advantage of Snakes algorithm could provide a methodology to impose geometric constraints while actively interacts to image data. This paper uses this idea of the Snakes algorithm to segment object ID characters in images of automation system. The printing area is 122 mm × 193 mm. The text should be justified to occupy the full line width, so that the right margin is not ragged, with words hyphenated as appropriate. Please fill pages so that the length of the text is no less than 180 mm.

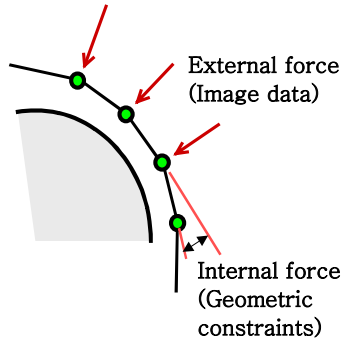


Fig. 2. Snakes algorithm: Advantage of Snakes algorithm could provide a methodology to impose geometric constraints while actively interacts to image data.

2.2 Multiple Templates

Suppose that it is impossible to extract the character regions from background of the inspection image if we use conventional binarization technique. In this section, we suggest a method to introduce multiple templates to find the character regions.

We form model templates from all characters included in the inspection images. Because all characters in input image are included in the template category, each pattern in the image corresponds to at least one character in this category. Searching the character area with these templates is to segment position of characters. That is, it is possible to search character area and recognize the character using template matching. However, in the case of poor quality images, this matching by single character template is also difficult and not reliable. Matching failure at preprocessing

stage leads to recognition failure at post-processing of optical character recognition (OCR) algorithm.

In order to solve this problem, we propose multiple templates, which is composed of N multiple templates. In target object of automation process, ID marks are usually not single character and a few characters are successively written on object surface to be inspected. Multiple templates present the model that may include several characters instead of one template in a template model. Multiple templates increase the possibility of matching because correlation value could be increased N times than that of single template. Fig. 3 shows the multiple template models. In designing multiple templates, there are some restrictions as follows.

(1) The size of each template must be same, if else, the size should be normalized to one size.

(2) No rotation and scale change in the character in input image. We only search the translation position the characters lie on.

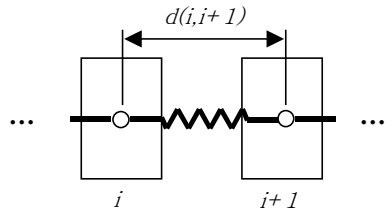


Fig. 3. Schematic of multiple templates model

Because multiple templates have to be successively connected, the height of each model template must be same for all model templates. In order to satisfy such condition, we normalized the size of template and treated matching with the pattern normalized from MBR (minimum boundary rectangle) of each template model.

Assume that object ID marks consist of N characters in inspection image. Hence, we should prepare the multiple templates with successive N characters. The number of total template model is M. Serial chain of multiple templates is configured for NGC matching using the prepared set of M model templates. The number of serial chain of multiple templates is same as the number of the characters in target image. Fig. 3 shows only two characters among whole serial chain. The M resulting values of NGC matching at i-th position are generated at neighboring locations of i-th position. The M resulting values of NGC matching at (i+1)-th position are also generated at neighboring locations of (i+1)-th position. Therefore, the number of combination of distance from matching positions generated by two successive templates is M^2. Because the numbers of serial chain are N, all possible path is M^{N-1} and we can define an optimal path search problem.

2.3 Modeling of ATM

The Snake paradigm models a deformable contour as possessing internal energy in order to impart smoothness to the contour. When this contour is located on an exter-

nal energy field, the contour seeks a local minimum of the energy field by moving and changing shape. The ATM is a modified version of the conventional dynamic programming method [1] for Snakes algorithm.

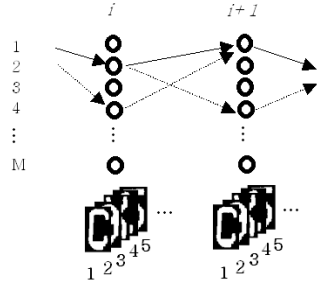


Fig. 4. Path search for character segmentation from dynamic programming

Fig.4 shows a concept diagram of path search using dynamic programming to segment character regions in poor quality images. First, assume that successive chain of characters is continuously positioned in any horizontal line in inspection image. The distance between two neighboring characters is not uniform and changeable. Then, full chain of templates try to find position of the N characters by overlapped to the horizontal line of the inspection image. Each node in i-th column of Fig. 4 presents optimally matched position at i-th stage of the N characters chain of templates. In each stage, there exist M node points because character category consists of M characters. The horizontal distance between two successive characters is changeable according to selection of any two linking nodes as the arrows of Fig. 4 present.

We try to find an optimal configuration increasing NGC coefficients while the distance between two neighboring templates maintains a predefined value d . The value d is mean distance between two successive characters calculated from several object ID marks in inspection images. This term as the geometric constraint corresponds to internal energy of Snakes algorithm and enforces pulling or compressing spring force when there is any difference from the mean distance. The horizontal distance for any two nodes between i-th and i+1-th stage defines an internal energy of Snakes model as follows

$$E^{i}_{\text{internal}} = \frac{1}{2}\alpha \cdot (d(v_i, v_{i-1}) - \bar{d})^2 . \tag{3}$$

The external energy of eq. (4) presents the similarity ratio from the template matching by NGC algorithm. This energy describes the degree of pulling strength of image data for each template.

$$E^{i}_{\text{external}} = \frac{1}{2}\beta \cdot Corr(v_{i-1}) \tag{4}$$

$$E_{\text{ATM}} = \sum_{i=1}^{N} (E^{i}_{\text{internal}} + E^{i}_{\text{external}}) \cdot \tag{5}$$

Therefore, the ATM energy is defined as eq. (5) and minimized by the following recursive dynamic programming algorithm:

$$S(n,m) = \min_{k}\{S(n-1,k) + E_{\mathrm{ATM}}(v_{n,m}, v_{n-1,k})\} \tag{6}$$

$$B(n,m) = k^{\min} \tag{7}$$

Where $S(n,m)$ represents the accumulated minimal energy level and the back pointer $B(n,m)$ holds the index $k(k = 1,...,M)$ giving minimum accumulation in each m step. After all stages have been processed, an optimal path is obtained by tracing back the pointers, beginning with the candidate that has a minimal $S(N,m)$ value. The time cost of the algorithm is a lower order $O(N-1\cdot M^{\square})$, when compared to the complexity $O(N-1\cdot M^{\square})$ of the conventional Snakes algorithm using DP search technique.

3 Experimental Results

In order to verify the usefulness of the proposed segmentation algorithm, we performed several experiments with poor quality images such as character images acquired from the surface of glass panel marked by laser unit and integrated circuit (IC) wafers of semiconductor process.

In case of frontal laser marks on the glass panel, it is engraved by melting the glass surface with laser optical unit. The rough area written by laser is located at a side of the glass panel and includes characters to be identified. Since the surface is rough by laser melting, the character area could provide bright visualization by diffuse reflection effects. Also the character area is located on the surface of the transparent glass, the visualization is poor as the light permeates. In Fig. 5(a), there is an example of laser mark images engraved on CRT glass panel. It is not easy to obtain clear visualization of the character area because of transparent property of the glass. Fig. 5(b) shows an IC wafer image obtained from semiconductor process. Wafers have the characteristics that the surface is reflective and the patterns on the background are complex and very noisy. When the image quality is sensitive to the variation of inspection environment and lighting condition, the performance in conventional systems is seriously degraded.

We need template images to implement the proposed ATM algorithm. The examples of the templates are shown in Fig. 6 and the models normalized to size from original templates are used to find the frontal laser marks on the glass panel. The all templates normalized to size have same height and different width.

Fig. 7 shows the experimental result to divide the character regions with the single model template. In case of using single template, the segmentation of unknown character is not reliable and fails in many cases because of loss of character properties from poor visualization and noises in the inspection image. In other words, we cannot match and segment a specific character by template matching if we use one template indicating a character. For example, the template indicating the character “C” always matches and segments the mark “C” of last location among six ID marks in the input image, as shown in Fig. 7. Hence, we cannot separate the first character mark “C” in the image. The template “0” falsely matches to the character “5” in the input image and we cannot segment ID character “0” in the image. Also, we previously don’t know what characters exist in current inspection image and thus, all templates in the

category of reference patterns should be tried to match the ID marks of the input image and this makes inevitable and frequent false matches.

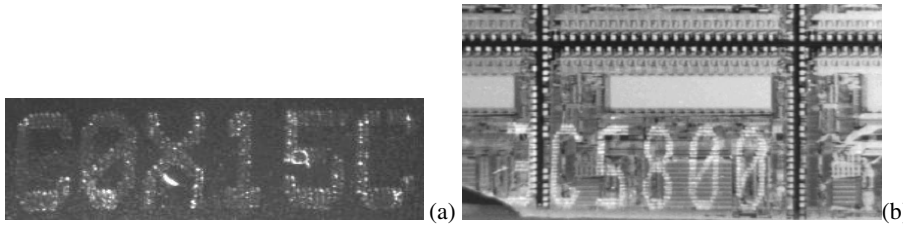


Fig. 5. Examples of poor quality images. (a) Frontal laser marks on CRT glass panel; (b) IC wafer image from semiconductor process.

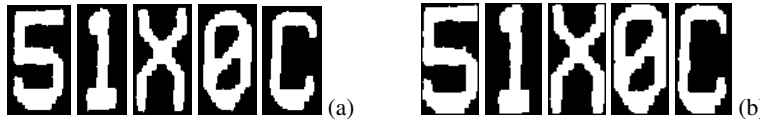


Fig. 6. Template images for analysis of frontal laser marks on CRT glass panel. (a) ID mark templates; (b) Size normalization of the original templates.

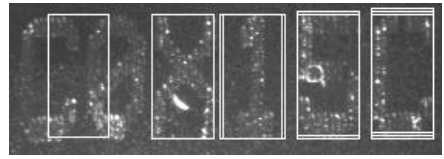


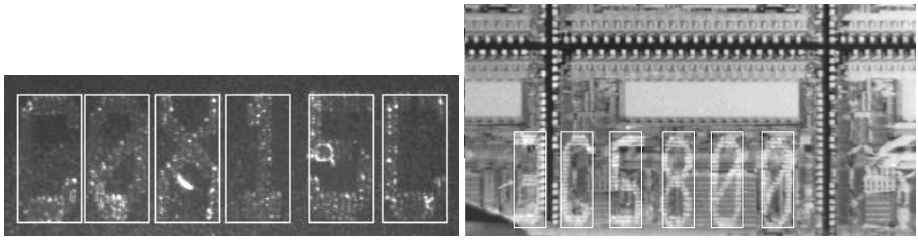
Fig. 7. Character segmentation by single template model

Fig. 8 shows the results of the character ID marks segmentation when we apply the proposed ATM method for a few images. Since these poor quality images have incomplete visualization, it is difficult to perform segmentation of the interesting character area with conventional methods. With a few additional examples, we proved the proposed methods well split unknown characters under bad visualization.

Because the time complexity of dynamic programming is proportional to the number of the model template in reference pattern category, the computational cost of ATM increases with the number of ID marks to be used. In the usual cases, template category of ID marks in FA processes is not large. In the experiment of Fig. 8(a), we use 10 model templates as the number of total reference pattern. We set the length of chain of ATM to 6-templates because the number of total ID marks in inspection image is six. Segmentation speed of the ATM requires about 1 sec in Pentium IV-2Ghz. Fig. 8(b) is another example from IC wafer manufacturing process. In this example, we used other reference patterns as model templates.

Segmentation result of ATM for Fig. 8(b) is similar to Fig. 8(a). Because search region in the image is broader than CRT panel image, computational cost of ATM is higher than that of CRT image. To implement the practical segmentation algorithm to

be applied in FA system, we should reduce the computational cost and any sensing and triggering techniques could restrict the search region of the input image.



(a) (b)

Fig. 8. Segmentation results by ATM. (a) Good visualization image of the CRT glass panel (b) Wafer image.

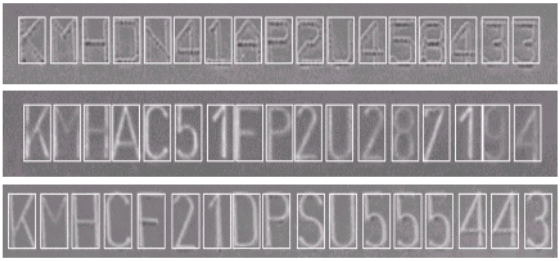


Fig. 9. Experiments for inspection process of the vehicle identification number in vehicle manufacturing factory.

Fig. 9 shows other experimental results that segment vehicle identification number(VIN) in automated vehicle manufacturing process. Almost commercial machine vision systems fail to segment the characters in these images. The proposed ATM algorithm could achieve the successful character segmentation in the poor quality images as shown in Fig. 9.

Through many additional experiments, we can identify the performance of ATM algorithm. The algorithm can provide a good pre-processing module to segment character regions in many OCR applications handling poor quality images.

Once we could segment character regions, then its possible to apply the character recognition algorithm such as artificial neural network or other graph search.

4 Conclusions

In this paper, we proposed an algorithm to solve the problem of the object ID marks segmentation for poor quality images of FA processes. Main contribution of this paper is a modeling of the functional modified from conventional Snakes algorithm to solve the character segmentation problem.

The key idea of the proposed methods is based on the assumption that under poor quality images, each single ID mark does not sufficiently denote features of the character area, but if the features of the character area are shown iteratively and continuously in any horizontal line of the inspection image, the areas have high possibility to be the character region that includes several characters. Therefore, the multiple templates are introduced and applied to discriminate character regions in the images including object ID marks. We proposed the ATM algorithm as the elastic multiple templates model that includes the spring between two neighboring templates. Each template of the multiple templates interacts to image data through the NGC-based pattern matching while the distance between two neighboring templates maintains a predefined mean distance. We define an optimization problem to solve the ATM. Dynamic programming as a search tool is introduced to reduce computational cost in optimization of the defined functional. ATM algorithm successfully finds and segments the object ID marks in several bad visualization images obtained from conventional FA processes.

References

1. Amini, A., Weymouth, T.E., Jain, R.C.: Using dynamic programming for solving variational problems in vision. IEEE Trans. Pattern Analysis and Machine Intelligence. 12 (1991)
2. Rosenfeld, A., Kak, A.C.: Digital picture processing. Academic Press New York (1976)
3. Chow, C. K., Kaneko, T.: Boundary detection of radiographic images by a thresholding method. Frontiers of Pattern Recognition, Academic Press New York (1972) 61-82
4. DongJong, K.: A fast and stable algorithm for medical images. Pattern Recognition Letters. 20 (1999) 507-512
5. Nagy, G., Kanai, J., Krishnamoorty, M., Thomas, M., Viswanathan, M.: Two complementary techniques for digitized document analysis. Proceedings of the ACM conference on Document Processing Systems, Santafe New Mexico (1988) 169-176
6. Aas, K.: Verification of Batch Numbers on Plastic Vials. Proceedings of Scandinavian Conference on Image Analysis (2001)
7. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active Contour Models. Int. J. of Computer Vision, Vol. 1. (1988) 321-331
8. Haralick, R., Shapiro, L.: Survey: Image segmentation technique. CVGIP 29 (1985) 100-132
9. Horowits, S., Pavlidis, T.: Picture segmentation by a tree traversal algorithm. J. CM. 23 (1976) 368-388
10. Lee, S. M.: Low rate video coding using 3-D segmentation with two change detection masks. ISO/IEC/JTC1/SC29/WG11 MPEG93/941 (1993)
11. Manickam, S., Roth, S. M., Bushman, T.: Intelligent and Optimal Normalized Correlation for High-Speed Pattern Matching. Datacube Technical Paper, Datacube Incorpolation (2000)

A Line-Based Pose Estimation Algorithm for 3-D Polyhedral Object Recognition

Tae-Jung Lho[1], Dong-Joong Kang[1], and Jong-Eun Ha[2]

[1] Mechatronics Engineering, Tongmyong University of Information Technology, 535, Yongdang-dong, Nam-gu, Busan, Korea
{tjlho, djkang}@tit.ac.kr

[2] Multimedia Engineering, Tongmyong University of Information Technology, 535, Yongdang-dong, Nam-gu, Busan, Korea
jeha@tit.ac.kr

Abstract. In this paper, we present a new approach to solve the problem of estimating the camera 3-D location and orientation from a matched set of 3-D model and 2-D image features. An iterative least-square method is used to solve both rotation and translation simultaneously. Because conventional methods that solved for rotation first and then translation do not provide good solutions, we derive an error equation using roll-pitch-yaw angle to present the rotation matrix. From the modeling of the error equation, we analytically extract the partial derivates for estimation parameters from the nonlinear error equation. To minimize the error equation, Levenberg-Marquardt algorithm is introduced with uniform sampling strategy of rotation space to avoid stuck in local minimum. Experimental results using real images are presented.

1 Introduction

This paper describes an algorithm that provides new solution for the problem of estimating camera location and orientation for pose determination from a set of recognized lines appearing in the image. In computer vision and related applications, we often wish to find objects based on stored models from an image containing objects of interest [1,6-7]. To achieve this, a model-based object recognition system first extracts sets of features from the scene and the model, and then it looks for matches between members of the respective sets. The hypothesized matches are then verified through pose estimation and model alignment, and possibly extended to be useful in various applications. Verification can be accomplished by hypothesizing enough matches to constrain the geometrical transformation from a 3-D model to a 2-D image under perspective projection.

If the correspondence is given from the relation between 3-D model lines and 2-D lines found in image, the goal of the 3-D object recognition is to find the rotation and translation matrices that maps the world coordinate system to the camera coordinate system. There have been several approaches to solve these problems. For more de-

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 906–914, 2004.
© Springer-Verlag Berlin Heidelberg 2004

tailed review, refer to Kumar [5]. The proposed 3-D pose algorithm in this paper is an extension of Ishii method [4] for 3-D pose decision. Ishii proposed a point correspondence method to decide camera parameters and pose detection. We extended the algorithm to line correspondence from the point algorithm.

Because using roll-pitch-yaw angle to present rotation is very simple and intuitive, we derive an error equation based on the roll-pitch-yaw angles as a nonlinear function. The error equation minimizes a point-to-plane distance that is defined as the dot product of unit normal to the projected plane of image line and a 3-D point on the model line transformed to camera coordinates. We directly extract the partial derivates for the estimation parameters from the nonlinear error equation.

The least-squares techniques to minimize the nonlinear function, are iterative in nature, and require an initial estimate. Because it is very difficult to provide good initial estimates avoiding local minimums without any additional information about object pose, we propose a simple uniform sampling strategy where the values uniformly sampled from three rotation angles of roll-pitch-yaw are provided to the initial estimates.

To handle the correspondence problem between 3-D model lines and 2-D image lines, we first extract junctions formed by two lines in the input image, and then find an optimal relation between the extracted junctions, by comparing them with previously constructed model relations. Junction detection acts as a line filter to extract salient line groups in the input image and then the relations between the extracted groups are searched to form a more complex group in an energy minimization framework. For more details, refer to Kang [2]. After solving the correspondence between two respective line sets, Levenberg-Marquardt technique is applied to minimize the defined error function. Partial derivatives for the error equation are analytically derived to form Jacobian matrices providing linearized form of the non-linear equation. Experimental results using real image of a 3-D polyhedral object are presented.

2 Geometric Constraints

Perspective projection between a camera and 3-dimentional objects defines a plane in 3-D space that is formed from a line in an image and the focal point of the camera. If no errors with perfect 3-D models were introduced in during image feature extraction, then model lines in 3-D space projecting onto this line in the image would exactly lie in this plane. This observation is the basis for the fit measure used in a few previous works [3-5]. We propose a pose decision method solving rigid transformation, which minimizes the sum-of-squared distances between points on 3-D model lines and the plane.

First of all, we describe the relationship between two coordinate systems. Camera and world coordinate systems are $O_c - xyz$, $O_w - XYZ$, respectively. Fig. 1 shows the two coordinate systems where X_w, Y_w, Z_w and x_C, y_C, z_C represent the axes of the

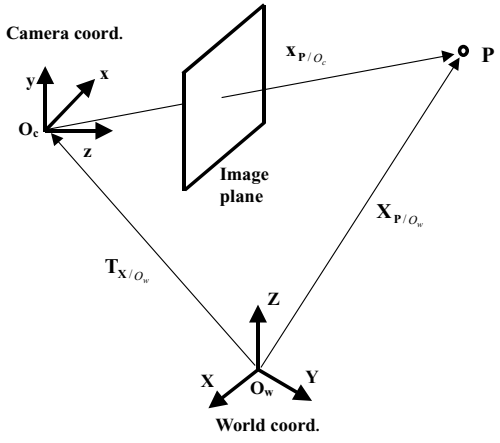


Fig. 1. The relationship between two coordinate systems.

world and camera coordinates system, respectively. The relationship between the two coordinate systems is given by the following vector-matrix equation:

$$\begin{pmatrix} x_{P/O_c} \\ y_{P/O_c} \\ z_{P/O_c} \end{pmatrix} = \mathbf{R}^t \cdot \begin{pmatrix} X_{P/O_W} - t_{1/O_W} \\ Y_{P/O_W} - t_{2/O_W} \\ Z_{p/O_W} - t_{3/O_W} \end{pmatrix}, \tag{1}$$

where $\mathbf{R}$ is rotation matrix from the world coordinate system to the camera coordinate system and $\mathbf{T} = (t_1 \ t_2 \ t_3)^t$ is translation vector from O_w point to O_c point. Upper index t presents transpose of a matrix. A point in 3-D space is represented by 3-D vector $\mathbf{P}$. We could use roll-pitch-yaw angles to represent the rotation matrix $\mathbf{R}$.

Each of the three rotations takes place about an axis in the fixed reference frame [3]:

$$\begin{aligned} \mathbf{R} &= \mathbf{R}_Z(\gamma)\mathbf{R}_Y(\beta)\mathbf{R}_X(\alpha) \\ &= \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix} \end{aligned} \tag{2}$$

Due to noise for extracting image lines, the segments usually will not lie exactly in the projection plane as shown in Fig. 2. The point-to-plane distance may be defined as the dot product of the unit normal to the plane and a 3-D point on the model line transformed to camera coordinates. It is possible to provide an error equation denoting the sum of squared perpendicular distances for line segments:

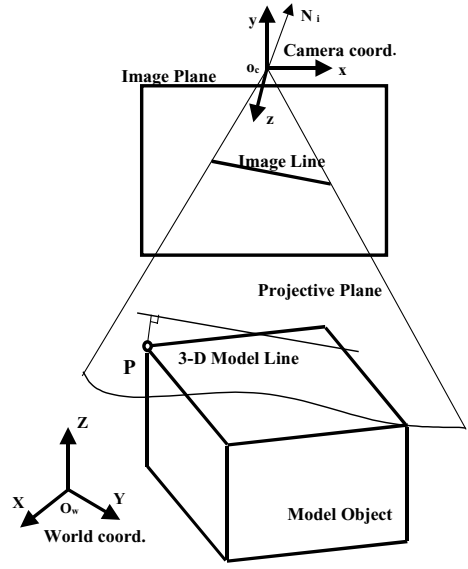


Fig. 2. The perpendicular distance by point-to-plane fit measure.

$$e = \sum\sum e_{ij}^2 = \sum_{i=1}^{l}\sum_{j=1}^{m}(\mathbf{N}_i \cdot (\mathbf{R}^t(\mathbf{P}_{ij} - \mathbf{T}))^2 \cdot \tag{3}$$

The summation is over l pairs of corresponding 3-D and 2-D line segments. A point $\mathbf{P}$ on the 3-D model line in Fig. 2 might be one among two endpoints and center point of the line. The index m is the number of points selected on the 3-D line. $\mathbf{N}_i$ is the unit normal vector to the plane formed by each 2-D segment. The pose of the 3-D segments relative to 2-D segments is expressed as a rotation $\mathbf{R}$ and translation $\mathbf{T}$ applied to the 3-D points. The best-fit 3-D pose for a set of corresponding 3-D and 2-D line segments is defined by the rotation $\mathbf{R}^*$ and translation $\mathbf{T}^*$ which minimize eq. (3). Solving for $\mathbf{R}^*$ and $\mathbf{T}^*$ is a non-linear optimization problem. We transpose the rotation matrix as follow:

$$\begin{aligned}\mathbf{R}^t &= \begin{pmatrix}1 & 0 & 0\\ 0 & \cos\alpha & \sin\alpha\\ 0 & -\sin\alpha & \cos\alpha\end{pmatrix}\begin{pmatrix}\cos\beta & 0 & -\sin\beta\\ 0 & 1 & 0\\ \sin\beta & 0 & \cos\beta\end{pmatrix}\begin{pmatrix}\cos\gamma & \sin\gamma & 0\\ -\sin\gamma & \cos\gamma & 0\\ 0 & 0 & 1\end{pmatrix}\\ &= \begin{pmatrix}r_{11} & r_{12} & r_{13}\\ r_{21} & r_{22} & r_{23}\\ r_{31} & r_{32} & r_{33}\end{pmatrix}^t\end{aligned} \tag{4}$$

and thus eq. (3) for a specific point P on the 3-D model line is rewritten as:

$$e_{ij} = \left(\begin{pmatrix}n_1 & n_2 & n_3\end{pmatrix}_i \cdot \begin{pmatrix}r_{11} & r_{12} & r_{13}\\ r_{21} & r_{22} & r_{23}\\ r_{31} & r_{32} & r_{33}\end{pmatrix}\begin{pmatrix}X_{ij} - t_1\\ Y_{ij} - t_2\\ Z_{ij} - t_3\end{pmatrix}\right) \tag{5}$$

where $\mathbf{n}_i$ is normal vector components of the plane obtained from a corresponding 2-D image line and 3-D model line. The $\mathbf{P}_{ij}$ in relative to world coordinate system is $(X_{ij}\ Y_{ij}\ Z_{ij})^t$ and translation vector $\mathbf{T}$ between two coordinate systems from O_w to O_c is given as $(t_1\ t_2\ t_3)^t$. The number of unknown parameters in eq. (5) is six for both $(t_1\ t_2\ t_3)$ of translation and $(\alpha\ \beta\ \gamma)$ of rotation.

From eq. (5), we can create an equation that expresses this error as the sum of the products of its partial derivatives:

$$\frac{\partial e}{\partial t_1}\delta t_1 + \frac{\partial e}{\partial t_2}\delta t_2 + \frac{\partial e}{\partial t_3}\delta t_3 + \frac{\partial e}{\partial \alpha}\delta\alpha + \frac{\partial e}{\partial \beta}\delta\beta + \frac{\partial e}{\partial \gamma}\delta\gamma = \delta e \,. \tag{6}$$

For example, we can obtain six equations for three lines using two end points of a line and hence produce a complete linear system, which can be solved for all six camera-model corrections. In conventional cases, several line segments could give an over-constrained linear equation. Levenberg-Marquardt method provides a solution for linearized form of the non-linear equation [8]. The small displacement vector $\delta\mathbf{x}$ including δt_1, δt_2, δt_3, $\delta\alpha$, $\delta\beta$, and $\delta\gamma$ represent errors of each parameter and define the Jacobian matrix $\mathbf{J}$. The partial derivatives of e with respect to each of the six parameters are calculated from eq. (5).

$$\frac{\partial e}{\partial t_1} = -(n_1 r_{11} + n_2 r_{12} + n_3 r_{13}) \,, \tag{7a}$$

$$\frac{\partial e}{\partial t_2} = -(n_1 r_{21} + n_2 r_{22} + n_3 r_{32}) \,, \tag{7b}$$

$$\frac{\partial e}{\partial t_3} = -(n_1 r_{31} + n_2 r_{32} + n_3 r_{33}) \,, \tag{7c}$$

$$\begin{aligned}\frac{\partial e}{\partial \alpha} &= n_1[(X_{ij}-t_1)\frac{\partial r_{11}}{\partial \alpha} + (Y_{ij}-t_2)\frac{\partial r_{21}}{\partial \alpha} + (Z_{ij}-t_3)\frac{\partial r_{31}}{\partial \alpha}] \\ &+ n_2[(X_{ij}-t_1)\frac{\partial r_{12}}{\partial \alpha} + (Y_{ij}-t_2)\frac{\partial r_{22}}{\partial \alpha} + (Z_{ij}-t_3)\frac{\partial r_{32}}{\partial \alpha}] \\ &+ n_3[(X_{ij}-t_1)\frac{\partial r_{13}}{\partial \alpha} + (Y_{ij}-t_2)\frac{\partial r_{23}}{\partial \alpha} + (Z_{ij}-t_3)\frac{\partial r_{33}}{\partial \alpha}]\end{aligned} \tag{8a}$$

$$\begin{aligned}\frac{\partial e}{\partial \beta} &= n_1[(X_{ij}-t_1)\frac{\partial r_{11}}{\partial \beta} + (Y_{ij}-t_2)\frac{\partial r_{21}}{\partial \beta} + (Z_{ij}-t_3)\frac{\partial r_{31}}{\partial \beta}] \\ &+ n_2[(X_{ij}-t_1)\frac{\partial r_{12}}{\partial \beta} + (Y_{ij}-t_2)\frac{\partial r_{22}}{\partial \beta} + (Z_{ij}-t_3)\frac{\partial r_{32}}{\partial \beta}] \\ &+ n_3[(X_{ij}-t_1)\frac{\partial r_{13}}{\partial \beta} + (Y_{ij}-t_2)\frac{\partial r_{23}}{\partial \beta} + (Z_{ij}-t_3)\frac{\partial r_{33}}{\partial \beta}]\end{aligned} \tag{8b}$$

$$\begin{aligned}\frac{\partial e}{\partial \gamma} &= n_1[(X_{ij}-t_1)\frac{\partial r_{11}}{\partial \gamma} + (Y_{ij}-t_2)\frac{\partial r_{21}}{\partial \gamma} + (Z_{ij}-t_3)\frac{\partial r_{31}}{\partial \gamma}] \\ &+ n_2[(X_{ij}-t_1)\frac{\partial r_{12}}{\partial \gamma} + (Y_{ij}-t_2)\frac{\partial r_{22}}{\partial \gamma} + (Z_{ij}-t_3)\frac{\partial r_{32}}{\partial \gamma}] \\ &+ n_3[(X_{ij}-t_1)\frac{\partial r_{13}}{\partial \gamma} + (Y_{ij}-t_2)\frac{\partial r_{23}}{\partial \gamma} + (Z_{ij}-t_3)\frac{\partial r_{33}}{\partial \gamma}]\end{aligned} \tag{8c}$$

Therefore, Jacobian matrix is

$$\mathbf{J} = \left[\frac{\partial e}{\partial \alpha} \; \frac{\partial e}{\partial \beta} \; \frac{\partial e}{\partial \gamma} \; \frac{\partial e}{\partial t_1} \; \frac{\partial e}{\partial t_2} \; \frac{\partial e}{\partial t_3} \right]^t . \tag{9}$$

The Levenberg-Marquardt method is an iterative variation on the Newton method in non-linear estimation. The normal equations $\mathbf{H}\delta\mathbf{x} = \mathbf{J}^t\mathbf{J}\delta\mathbf{x} = \mathbf{J}^t\mathbf{e}$ are augmented to $\mathbf{H}'\delta\mathbf{x} = \mathbf{J}^t\mathbf{e}$ where $\mathbf{H}' = (1+\lambda\mathbf{I})\mathbf{H}$. The value λ is initialized to a small value. If the value obtained for $\delta\mathbf{x}$ reduces the error, the increment of $\mathbf{x}$ to $\mathbf{x}+\delta\mathbf{x}$ is accepted and λ is divided by 10 before the next iteration. On the other hand, if the error increases, then λ is multiplied by 10 and the augmented normal equations are solved again, until an increment that reduces the error is obtained.

3 Experiments

Experiments show that there is rapid convergence even with significant errors in the initial estimates. Several iterations are enough to obtain convergence of the parameters.

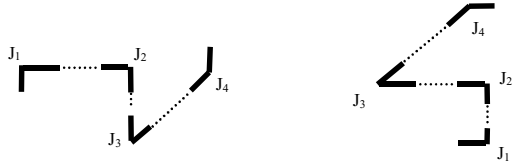


Fig. 3. Topological shapes for an object model; (a) A model description that consists of junction combination with clockwise rotational direction; (b) Another model description with counter-clockwise direction.

Fig. 4 presents an example extracting a topological line groups to guide 3-D object recognition. The method applied to find the correspondences between model and scene line features is described in Kang [2]. The topological shapes are invariant to the wide change of view variations. If there is no self-occlusion on the object, the interesting line groups can be possibly extracted. Fig. 4(a) shows the original image to be tested. After discarding the shorter lines, Fig. 4(b) presents the extracted lines with the numbering indicating the line index, and Fig. 4(c) gives the matched line groups corresponding to the model shape of Fig. 3(a) and 3(b), respectively. In each extraction, there are enough line groups to guide a hypothesis for 3-D object recognition.

Fig. 5 shows the model overlapped from the 3-D pose determination algorithm of Section 2. As shown in Fig. 5, the lines matched in Fig. 4 are enough to guide an initial hypothesis for 3-D object recognition. In this experiment, we set m=3 corresponding to two endpoints and a center point for a model line. And arbitrary three translation values and uniformly sampled rotation values in $\alpha-\beta-\gamma$ angle space are

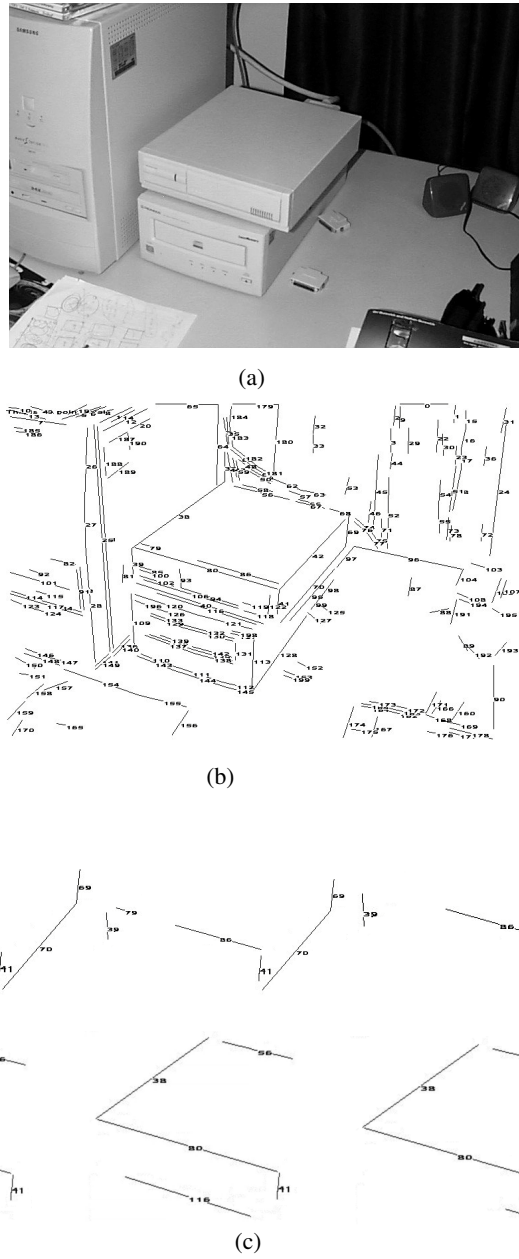


Fig. 4. Topological shape extraction for 3-D object recogntion. (a) Original image; (b) Line extraction; (c) Found topological shapes.

selected for initial values of six-pose parameters. In a few iteration steps, the convergence is reached. If the error function (3) for any initial values of six pose parameters

is not reduced during a few iterations, the initial candidates are discarded and next initial values are tried from uniformly sampled angles. From several experiments of convergence starting from many random translation parameters with sampled angles, we can conclude the stable convergence if there exists a solution.
For all correspondences in Fig. 4(c), we can obtain stable convergence as shown in Fig. 5.

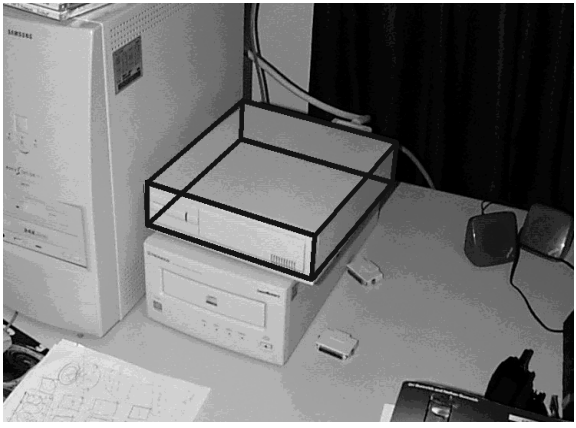


Fig. 5. Model is overlapped to real image from the hypotheses in Fig. 4

4 Conclusions

In computer vision and related applications, a model-based object recognition system first extracts sets of features from the scene and the model, and then it looks for matches between members of the respective feature sets. The hypothesized matches are then verified through pose estimation and model alignment. Verification can be accomplished by hypothesizing enough matches to constrain the geometrical transformation from a 3-D model to a 2-D image under perspective projection. This paper presents a new method to estimate the camera 3-D location and orientation from a matched set of 3-D model and 2-D image lines. If the correspondence is given from the relation between 3D model lines and 2D lines found in image, a key step of the 3-D object recognition is to find the rotation and translation matrices that map the world coordinate system to the camera coordinate system.

This paper proposes a method using roll-pitch-yaw angle to present rotation. We derive a nonlinear error equation based on the roll-pitch-yaw angles. The error equation is designed to minimize a point-to-plane distance that is defined as the dot product of unit normal to the projected plane of image line and a 3-D point on the model line transformed to camera coordinates. Levenberg-Marquardt method minimizes the error equation with uniform sampling strategy of rotation space to avoid stuck in local minimum.

From experiments using real images, the proposed method is proved to be stable to initial values of estimating parameters. From corresponding line sets between 3-D model and 2-D real images, the method converses to good pose solutions in only a few iteration.

References

1. Lowe, D.G.: Three-Dimensional Object Recognition from Single Two-Dimensional Images. Artificial Intelligence 31 (1987) 355-395
2. Kang, D.J., Ha, J.E., Kweon, I.S.: Fast Object Recognition using Dynamic Programming from Combination of Salient Line Groups. Pattern Recognition, Vol.36 (2003) 79-90
3. Craig, J.J.: Introduction to Robotics: Mechanics and Control. 2nd Ed., Addison-Wesley Publishing (1989)
4. Ishii, M., Sakane, S., Kakikura, M., Mikami, Y.: A 3-D Sensor system for Teaching Robot Paths and Environments. Int. J. Robotics Research, Vol.6 (1987) 45-59
5. Kumar, R. Hanson, A.R.: Robust methods for Estimating Pose and a Sensitivity Analysis. CVGIP: Image Understanding, Vol. 60 (1994) 313-342
6. Li, S.Z.: Matching: invariant to translations, rotations and scale changes. Pattern Recognition, Vol. 25. (1992) 583-594
7. Grimson, W.E.L., Lozano-Perez, T.: Localizing overlapping parts by searching the interpretation tree. IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 9. (1987) 469-482
8. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes in C, Cambridge Press (1992)

Initialization Method for the Self-Calibration Using Minimal Two Images

Jong-Eun Ha[1] and Dong-Joong Kang[2]

[1] Multimedia Engineering, Tongmyong University of Information Technology,
535, Yongdang-dong, Nam-gu, Busan, Korea
jeha@tit.ac.kr
[2] Mechatronics Engineering, Tongmyong University of Information Technology,
535, Yongdang-dong, Nam-gu, Busan, Korea
djkang@tit.ac.kr

Abstract. Recently, 3D structure recovery through self-calibration of camera has been actively researched. Traditional calibration algorithm requires known 3D coordinates of the control points while self-calibration only requires the corresponding points of images, thus it has more flexibility in real application. In general, self-calibration algorithm results in the nonlinear optimization problem using constraints from the intrinsic parameters of the camera. Thus, it requires initial value for the nonlinear minimization. Traditional approaches get the initial values assuming they have the same intrinsic parameters while they are dealing with the situation where the intrinsic parameters of the camera may change. In this paper, we propose new initialization method using the minimum 2 images. Proposed method is based on the assumption that the least violation of the camera's intrinsic parameter gives more stable initial value. Synthetic and real experiment shows this result.

1 Introduction

Recently, 3D structure recovery through self-calibration of camera has been actively researched. Traditional calibration algorithm requires known 3D coordinates of the control points while self-calibration only requires the corresponding points of images, thus it has more flexibility in real application. In general, self-calibration algorithm results in the nonlinear optimization problem using constraints from the intrinsic parameters of the camera. Thus, it requires initial value for the nonlinear minimization. Traditional approaches get the initial values assuming they have the same intrinsic parameters while they are dealing with the situation where the intrinsic parameters of the camera may change.

Faugeras et al. [1] proposed a self-calibration algorithm that uses the Kruppa equation. It enforces that the planes through two camera centers that are tangent to the absolute conic should also be tangent to both of its images. Hartley [2] proposed another method based on the minimization of the difference between the internal camera parameters for the different views. Polleyfeys et al. [3] proposed a stratified

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 915–923, 2004.
© Springer-Verlag Berlin Heidelberg 2004

approach that first recovers the affine geometry using the modulus constraint and then recover the Euclidean geometry through the absolute conic. Heyden & Astrom [4], Triggs [5] and Pollefeys & Van Gool [6] use explicit constraints that relate the absolute conic to its images. These formulations are especially interesting since they can easily be extended to deal with the varying internal camera parameters.

Recently self-calibration algorithms that can deal with the varying camera's intrinsic parameters were proposed. Heyden & Astrom [7] proposed a self-calibration algorithm that uses explicit constraints from the assumption of the intrinsic parameters of the camera. They proved that self-calibration is possible under varying cameras when the assumptions that the aspect ratio was known and no skew establishes about the camera.

They solved the problem using the bundle adjustment that requires simultaneous minimization on the all reconstructed points and cameras. Moreover, the initialization problem was not properly presented. Bougnoux [8] proposed a practical self-calibration algorithm that used the constraints derived from Heyden & Astrom [7]. He proposed the linear initialization step in the nonlinear minimization. He used the bundle adjustment in the projective reconstruction step. Similarly, Pollefeys et al. [9] proposed a versatile self-calibration method that can deal with a number of types of constraints about the camera. They showed a specialized version for the case where the focal length varies, possibly also the principal point.

In this paper, we propose new initialization method for the self-calibration algorithm by using the minimal two images, which result in the more stable initial values for the nonlinear minimization. This results in the solving of the simultaneous equations of the second-order, and gives two solutions that have opposite direction in projective space. Proposed method is based on the assumption that the least violation of the camera's intrinsic parameter gives more stable initial value. Real experiment shows this result.

2 Self-Calibration Algorithm Review

In this section, we review the self-calibration algorithm that appears in [8]. The process of projection of a point in 3D to the image plane can be represented as the following sequential steps:

$$\mathbf{P}_{euc} = \mathbf{A}\mathbf{P}_0\mathbf{T} = \begin{bmatrix} \alpha_u & \gamma & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}_3^T & 1 \end{bmatrix} \tag{1}$$

where $\mathbf{T}$ represents the transformation of coordinate systems from world to the camera-centered system, $\mathbf{P}_0$ is the perspective projection and $\mathbf{A}$ consists of the intrinsic parameters of camera. α_u and α_v are the scale aspect of the x and y axis in the image, γ is a skew factor, and u_0 and v_0 are the image coordinate of the principle point.

The following assumptions about the intrinsic parameters of camera are used in the self-calibration in [8].

$$\begin{aligned} \gamma &= 0 \\ \alpha_u &= \alpha_v \end{aligned} \tag{2}$$

It is well known that we can reconstruct a scene up to a projective transformation using only the corresponding points on the images [10,11]. This can be represented as:

$$\tilde{\mathbf{m}}_j^i \cong \mathbf{P}_{proj}^i \tilde{\mathbf{M}}_j^{proj} = \mathbf{P}_{proj}^i \mathbf{Q}\mathbf{Q}^{-1} \tilde{\mathbf{M}}_j^{proj} \tag{3}$$

where $\tilde{\mathbf{m}}_j^i$ is the j-th point in the i-th image, $\mathbf{P}_{proj}^i$ is a projective projection matrix of the i-th image and $\tilde{\mathbf{M}}_j^{proj}$ is a projective structure of scene point corresponding to the image point $\tilde{\mathbf{m}}_j^i$.

The projective structure $\tilde{\mathbf{M}}_j^{proj}$ is related to the metric structure by a projective transformation matrix $\mathbf{Q}$. In Eq. (3), any nonsingular matrix $\mathbf{Q}$ satisfies the above relation, so there can be many projective reconstructions. There exists a unique $\mathbf{Q}$ matrix that transforms the projective structure to a metric structure of a given scene. Finding this $\mathbf{Q}$ matrix is calibration process. We can obtain the Euclidean projection matrix and metric structure of a scene using this unique $\mathbf{Q}$ matrix.

$$\begin{aligned} \mathbf{P}_{euc}^i &\cong \mathbf{P}_{proj}^i \mathbf{Q} \\ \tilde{\mathbf{M}}_j^{euc} &\cong \mathbf{Q}^{-1} \tilde{\mathbf{M}}_j^{proj} \end{aligned} \tag{4}$$

In general, under the pinhole camera model, we can set the projective projection matrix of the first camera as $\mathbf{P}_{proj}^1 = [\mathbf{I}_3 \quad \mathbf{0}_3]$. We have the Euclidean projection matrix of the first camera $\mathbf{P}_{euc}^1 = [\mathbf{A}_1 \quad \mathbf{0}_3]$ if we set the world coordinate at the optical center of the first camera. If we substitute these projection matrices in Eq. (4), then we have

$$\begin{aligned} \mathbf{P}_{euc}^1 \cong \mathbf{P}_{proj}^1 \mathbf{Q} &\Leftrightarrow [\mathbf{A}_1 \quad \mathbf{0}_3] \cong [\mathbf{I} \quad \mathbf{0}_3]\mathbf{Q} \\ &\Leftrightarrow \exists(\mathbf{q}, q_{44}) \mid \mathbf{Q} \cong \begin{bmatrix} \mathbf{A}_1 & \mathbf{0}_3 \\ \mathbf{q}^T & q_{44} \end{bmatrix} \end{aligned} \tag{5}$$

Here, $\mathbf{Q}$ is defined up to a scale and it can be represented as

$$\mathbf{Q} = \begin{bmatrix} f & 0 & u_0 & 0 \\ 0 & f & v_0 & 0 \\ 0 & 0 & 1 & 0 \\ q_1 & q_2 & q_3 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{A}_1 & \mathbf{0}_3 \\ \mathbf{q}^T & 1 \end{bmatrix} \quad (f \equiv \alpha_u, \alpha_v) \tag{6}$$

Now $\mathbf{Q}$ matrix contains six unknowns. Next we review the constraint for obtaining $\mathbf{Q}$ matrix.

If we set $\mathbf{P}_{euc}^i$

$$\mathbf{P}^i_{euc} \cong \mathbf{P}^i_{proj}\mathbf{Q} = \begin{bmatrix} \mathbf{p}_1^{iT} & \\ \mathbf{p}_2^{iT} & \mathbf{t}_i \\ \mathbf{p}_3^{iT} & \end{bmatrix} \tag{7}$$

then we have the following constraint for the Euclidean projection matrix, $\mathbf{P}_{euc}$.

$$\begin{aligned} &\gamma = 0 \Leftrightarrow (\mathbf{p}_1 \times \mathbf{p}_3) \bullet (\mathbf{p}_2 \times \mathbf{p}_3) = 0 \\ &\alpha_u = \alpha_v \Leftrightarrow \left\| \mathbf{p}_1 \times \mathbf{p}_3 \right\| = \left\| \mathbf{p}_2 \times \mathbf{p}_3 \right\| \end{aligned} \tag{8}$$

These equations give two constraints for the unknown $\mathbf{Q}$ matrix for each camera and we can obtain the solution using at least 4 images. The resulting problem can be formulated as the nonlinear estimation that minimizes Eq. (8) for each camera.

3 Initialization Method and Detailed Procedures

3.1 Initialization Method Using Minimal Two Images

We need initial values to run the nonlinear minimization. The cost function of Eq. (8) has many local minima. Thus, it is important to have good initial values close to the true ones to guarantee the convergence. The initial value by [8] often does not guarantee convergence. This is due to the fact that least-squares solution under the assumption that intrinsic parameters are constant under the varying cameras makes the initial values even worse. We propose new initialization method for the $(q_1, q_2, q_3)^T$ using only two views, thus guarantee the least violation of the varying camera situation.

Euclidean projection matrix can be represented as:

$$\mathbf{P}^i_{euc} = \mathbf{A}_i \left[\mathbf{R}_i \, \mathbf{t}_i \right]$$

where $\mathbf{A}_i$ is the matrix consisted of the intrinsic parameters, $\mathbf{R}_i$ is rotation matrix and $\mathbf{t}_i$ is translation vector. Also if we denote $\mathbf{Q}^*$ as the first three column of the matrix $\mathbf{Q}$, we can derive following relations from Eq. (4).

$$\mathbf{A}_i \mathbf{R}_i \cong \mathbf{P}^i_{proj}\mathbf{Q}^* \tag{9}$$

From Eq. (9), it follows

$$\begin{aligned} &(\mathbf{A}_i \mathbf{R}_i)(\mathbf{A}_i \mathbf{R}_i)^T \cong \left(\mathbf{P}^i_{proj}\mathbf{Q}^*\right)\left(\mathbf{P}^i_{proj}\mathbf{Q}^*\right)^T \\ &\omega_k \cong \mathbf{P}^i_{proj} \Omega \, \mathbf{P}^{i\,T}_{proj} \qquad \left(\omega_k = \mathbf{A}_i \mathbf{A}_i^T, \; \Omega = \mathbf{Q}^* \mathbf{Q}^{*T} \right) \end{aligned} \tag{10}$$

ω_k is the dual image absolute conic and Ω is the absolute dual quadric.
From Eq. (10), we obtain

$$\lambda_i \begin{bmatrix} f_i^2+u_0^2 & u_0v_0 & u_0 \\ u_0v_0 & f_i^2+v_0^2 & v_0 \\ u_0 & v_0 & 1 \end{bmatrix} = \mathbf{P}_{proj}^{i} \begin{bmatrix} f_1^2+u_0^2 & u_0v_0 & u_0 & f_1q_1+u_0q_3 \\ u_0v_0 & f_1^2+v_0^2 & v_0 & f_1q_2+v_0q_3 \\ u_0 & v_0 & 1 & q_3 \\ f_1q_1+u_0q_3 & f_1q_2+v_0q_3 & q_3 & \|\mathbf{q}\|^2 \end{bmatrix} \mathbf{P}_{proj}^{iT}. \quad (11)$$

We use f computed from the algorithm in [8], and take the initial value of the principle point as the center of the first image. Then, we are left with four unknowns $\lambda_i, (q_1, q_2, q_3)^T$, and Eq. (10) provides 6 equations. We compute the unknowns using four equations. Thus, we avoid the false initial value by least-squares by overconstrained equations. Experimental results support this fact.

3.2 Additional Constraint on the Position of the Principal Point

We add additional constraint to improve the behavior of the algorithm in the nonlinear minimization [12]. We observed that the algorithm often gives some erroneous results when we only use the two constraints of Eq. (8). The accuracy of the algorithm depends on the accuracy of the projective reconstruction. We used the simple linear method of Hartley [13] for the projective reconstruction, which gives a comparable result to that of nonlinear minimization under the Gaussian noise distribution.

In spite of the small residual in the projective reconstruction, the algorithm often gives a false result as noise level increases while the residual of the nonlinear minimization decreased. This is due to the lack of the ability of the two constraints to constrain the solution space in the meaningful range when the noise perturbs the projective projection matrices. Each constraint is a 4th order polynomial so that the algorithm is very sensitive to the noise.

We partially overcome this problem by adding additional constraint on the minimization. This is from the experimental observations of the behavior of the algorithm that only uses the two constraints of Eq. (8). Among the six unknowns the principal point was most sensitive in the nonlinear minimization. Therefore it is necessary to constrain the position of principal point in a restricted area to obtain a meaningful solution. We add the following constraint on the principal point.

$$\begin{aligned} \tilde{u}_0 - \hat{u}_0 &= 0 \\ \tilde{v}_0 - \hat{v}_0 &= 0 \end{aligned} \quad (12)$$

where u_0, v_0 is the image center of the first camera and u_0, v_0 is the principal point computed from the decomposition of the estimated Euclidean projection matrix during the minimization process.

The object function for the minimization with additional constraint for the principle point is

$$\begin{aligned} E(f,u_0,v_0,q_1,q_2,q_3) = &\sum_{i=2}^{N} \left[\left(\mathbf{p}_1^i \times \mathbf{p}_3^i\right) \bullet \left(\mathbf{p}_2^i \times \mathbf{p}_3^i\right) \right]^2 + \sum_{i=2}^{N} \left[\left\|\mathbf{p}_1^i \times \mathbf{p}_3^i\right\| - \left\|\mathbf{p}_2^i \times \mathbf{p}_3^i\right\| \right]^2 \\ &+ \sum_{i=2}^{N} \left[(\tilde{u}_0 - \hat{u}_0)^2 + (\tilde{v}_0 - \hat{v}_0)^2 \right] \end{aligned} . \quad (13)$$

We do not fix the value of the principal point of the first camera as the known value of the image center of the first image. This gives also a bad behavior in the minimization process.

The overall structure of the self-calibration algorithms is as follows:

(a) Find the projection matrix $\mathbf{P}_{proj}^{i}$ by projective reconstruction between images, e.g., 1-2, 1-3, …, 1-N.
(b) Transform the projective reconstruction of (a) to have equal basis in P^3.
(c) Obtain an initial value of the unknowns using the linear method.
(d) Obtain a 4X4 homography Q through the nonlinear minimization.
(e) Recover the Euclidean projection by $\mathbf{P}_{euc}^{i} \cong \mathbf{P}_{proj}^{i}\mathbf{Q}$.
(f) Recover Euclidean structure by $\tilde{\mathbf{M}}_{j}^{euc} \cong \mathbf{Q}^{-1}\tilde{\mathbf{M}}_{j}^{proj}$

The step (b) is necessary because Eq. (3) establishes under an equal basis in P^3. We use the method from Csurka & Horaud [16] to transform the projective reconstruction to have equal basis in P^3.

It is necessary to know all the true value of unknown parameters to check the proposed algorithm using the synthetic image. Except the value of $\mathbf{q} = (q_1, q_2, q_3)^T$, all other values are easily obtained in the experiment using synthetic image.

One possible method is to set the value of $\mathbf{q} = (q_1, q_2, q_3)^T$ arbitrary, and then obtains the projective projection matrix using the relation $\mathbf{P}_{proj}^{i} \cong \mathbf{P}_{euc}^{i}\mathbf{Q}^{-1}$. But this procedure can't analyze the algorithm from the given correspondence. Projective projection matrix is obtained using the fundamental matrix, which is computed using the correspondence. Thus, other method for computing $\mathbf{q} = (q_1, q_2, q_3)^T$ is necessary to analyze proposed algorithm at every step. The method in [16], which computes the homography between Euclidean structure and projective structure, is used.

True fundamental matrix is computed from the known value of intrinsic and extrinsic parameters, then true projective projection matrices are computed. From these projective projection matrices, projective structure is computed using the correspondence without noise. Finally, $\mathbf{q} = (q_1, q_2, q_3)^T$ is computed from the homography between Euclidean structure and projective structure.

4 Experimental Results

Fig. 1 represents calibration box and control points used in the experiments, and they are acquired by a color CCD camera (Sony EVI-300). We changed internal parameters of camera during acquisition by varying zoom. Calibration box size is 150mm x 150mm x 150mm. Tsai [17], Bougnoux [8] and proposed algorithm is compared.

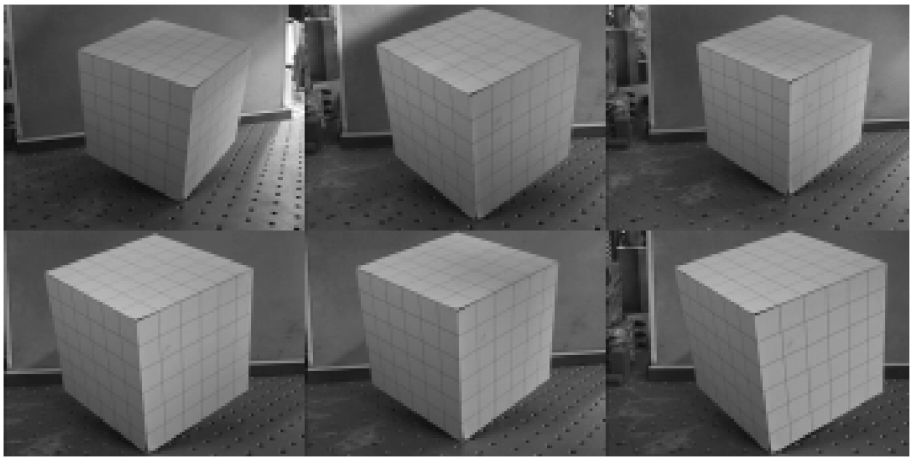


Fig. 1. Calibration box image sequences used in the experiments.

Table 1 shows the estimated initial f and $\mathbf{q} = (q_1, q_2, q_3)^t$. We can see that proposed method gives more accurate initial values compared to [8]. Table 2 shows the estimated intrinsic parameters of the camera. Proposed method use additional constraint about the principal point [12] and new initialization method. Table 3 and 4 shows estimated extrinsic parameters. Bougnoux [8] gives large departure from that of Tsai [17] while proposed algorithm gives comparable results.

Finally Fig. 2 shows the estimated 3D structure by each algorithm. The unknown scale is set using the first control points between Tsai [17] and other two methods. After fixing the unknown scale, 3D error by all control points compared to the Tsai's method is as follows:

Bougnoux [8]: (mean, std)=(83.1,80.9) [mm]
Proposed algorithm: (mean, std)=(5.91,6.56) [mm]

Table 1. Comparison of the estimated initial f and $\mathbf{q} = (q_1, q_2, q_3)^T$

true value	linear method [8]	proposed method
758.9,(-50.8,12.3,-36.6)	637.7,(-77.5,0.446,-6.00)	637.7,(-39.7,10.2,-35.1)

Table 2. Comparison of the estimated intrinsic parameters $(\alpha_u, \alpha_v, u_0, v_0)$

	Tsai [17]	linear method [8]	proposed method
C1	(758.9,760.0,340.4,230.3)	(197.4,197.4,253.7,229.5)	(605.6,605.6,283.0,237.3)
C2	(880.2,881.0,352.2,223.2)	(223.1,222.9,364.7,224.3)	(692.6,691.1,340.4,230.3)
C3	(859.9,861.2,366.6,220.8)	(206.4,206.5,339.3,227.6)	(666.5,666.7,332.9,232.0)
C4	(1092,1094.1,332.6,232.1)	(253.1,253.6,330.3,249.0)	(836.7,835.4,311.4,248.6)
C5	(1007,1007.6,335.4,234.7)	(239.3,239.1,303.7,242.3)	(777.1,775.6,298.6,247.4)
C6	(1032,1034.5,326.5,231.5)	(252.0,252.0,418.0,248.8)	(803.5,800.6,350.4,247.2)

Table 3. Comparison of the estimated rotation $(\theta_x, \theta_y, \theta_z)$ between camera i and j.

	Tsai [17]	linear method [8]	proposed method
1-2	(-1.03,-15.4,-4.07)	(-0.263,-3.97,-3.87)	(-0.869,-12.2,-4.10)
1-3	(1.51,-9.61,-2.66)	(0.372,-2.54,-2.85)	(1.12,-7.89,-2.84)
1-4	(1.04,-15.7,-3.25)	(0.265,-3.98,-3.43)	(0.747,-12.7,-3.46)
1-5	(1.47,-11.1,-1.94)	(0.375,-2.90, 2.15)	(1.13,-9.06,-2.14)
1-6	(0.895,-23.6,-4.39)	(0.233,-5.90,-4.53)	(0.661,-18.6,-4.62)

Table 4. Comparison of the estimated translation between camera i and j.

	Tsai [17]	linear method [8]	proposed method
1-2	(-0.970,0.070,0.234)	(-0.996,0.071,0.061)	(-0.980,0.068,0.189)
1-3	(-0.688,-0.149,0.710)	(-0.945,-0.217, 0.244)	(-0.753,-0.178, 0.633)
1-4	(-0.755,-0.034,0.655)	(-0.979,-0.055,0.197)	(-0.827,-0.047, 0.560)
1-5	(-0.740,-0.119,0.661)	(-0.964,-0.163,0.209)	(-0.808,-0.141,0.572)
1-6	(-0.864,-0.078,0.497)	(-0.987,-0.091,0.131)	(-0.912,-0.087,0.401)

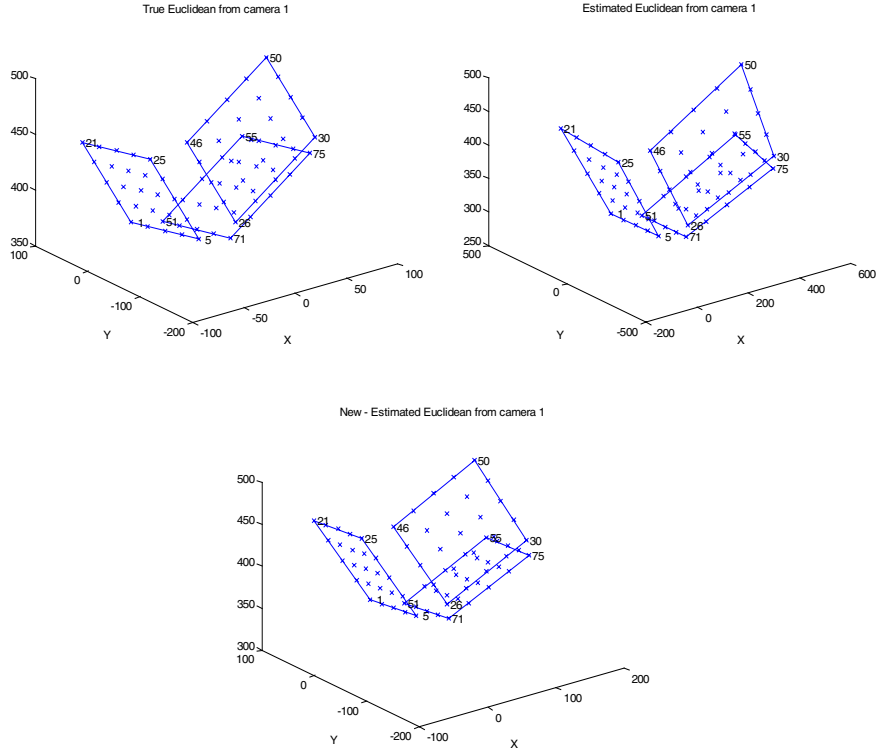


Fig. 2. Estimated 3D structure by Tsai[17], Bougnoux[8], and proposed algorithm.

5 Conclusion

New initialization method for the self-calibration using the minimum 2 views is presented. Proposed method is based on the assumption least violation about the camera gives more accurate initial values for the self-calibration. Experimental results using calibration box shows this fact. Future research will focus on the behavior of plane at infinity in the self-calibration.

References

1. Faugeras, O., Luong, Q.-T., Maybank, S.: Camera Calibration: Theory and experiments. European Conference on Computer Vision, (1992) 321-334
2. Hartley, R.: Euclidean reconstruction from uncalibrated views. Applications of invariance in Computer Vision, LNCS 825, Springer-Verlag (1994)
3. Pollefeys, R., Van Gool, R., Oosterlinck, A.:The Modulus Constraint: A New Constraint for Self-Calibration. International Conference on Pattern Recognition, (1996) 349-353
4. Heyden, A., Astrom, K.: Euclidean Reconstruction from Constant Intrinsic Parameters. International Conference on Pattern Recognition (1996)
5. Triggs, B.: The Absolute Quadric. Proc. Computer Vision and Pattern Recognition (1997)
6. Pollefeys, M., Van Gool, L.: Self-calibration from the absolute conic on the plane at infinity. Proc. CAIP'97 (1997)
7. Heyden, A., Astrom, K.: Euclidean Reconstruction from Image Sequences with Varying and Unknown Focal Length and Principal Point. Proc. CVPR (1997)
8. Bougnoux, S.: From Projective to Euclidean Space under any practical situation, a criticism of self-calibration. Proc. ICCV (1998) 790-796
9. Pollefeys, M., Koch, R., Van Gool, L.: Self-calibration and Metric Reconstruction in Spite of Varying and Unknown Internal Camera Parameters. Proc. ICCV. (1998) 90-95
10. Faugeras, O.: What can be seen in three dimensions with an uncalibrated stereo rig?, Proc. ECCV (1992) 563-578
11. Hartley, R., Gupta, R., Chang, T.: Stereo from uncalibrated cameras. Proc. CVPR, (1992) 761-764
12. Ha, J.E., Yang, J.Y., Yoon, K.,J., Kweon, I., S.: Self-calibration using the linear projective reconstruction. Proceedings of the International Conference on Robotics and Automation, (2000) 885-890
13. Hartley, R.: In defence of the 8-point algorithm. Fifth International Conference on Computer Vision, pp. 1064-1070, 1995.
14. Hartley, R., Sturm, P.: Triangulation. Computer Vision and Image Understanding, Vol. 68 (1997) 146-157
15. Rothwell, C., Faugeras, O., Csurka, G.: A comparison of projective reconstruction methods for pairs of views. Computer Vision and Image Understanding, Vol. 68 (1997) 36-58
16. Csurka, G., Horaud, R.: Finding the collineation between two projective reconstructions. INRIA RR-3468 (1998)
17. Tsai, R. Y.: A versatile camera calibration technique for high accuracy 3D machine vision metrology using off-the-self TV cameras and lenses. IEEE Journal of Robotics and Automation, Vol. 3 (1987)

Face Recognition for Expressive Face Images

Hyoun-Joo Go, Keun Chang Kwak, Sung-Suk Kim, and Myung-Geun Chun

School of Electrical and Computer Engineering, Chungbuk National University
Cheongju, Korea
mgchun@chungbuk.ac.kr

Abstract. In this paper, we deal with a face recognition method for the expressive face images. Since the face recognition is one of the most natural and straightforward biometric methods, there have been various research works. However, most of them are focused on the expressionless face images. In real situations, however, it is required to consider the emotional face images. Here, three basic human emotions such as happiness, sadness, and anger are investigated. The face recognition becomes a very difficult problem if we consider the facial expression. This situation requires a robust face recognition algorithm. So, we use a fuzzy linear discriminant (LDA) algorithm with the wavelet transform. The fuzzy LDA is a statistical method that maximizes the ratio of between-scatter matrix and within-scatter matrix and also handles the fuzzy class information.

1 Introduction

Biometrics is defined as the capture and use of physiological or behavioral characteristics for personal identification or individual verification purposes. Face recognition among biometrics is a natural and straightforward biometric method to identify each other. Face recognition has been researched in various areas. However, face recognition is a very difficult problem due to large variation in light direction, face pose, and facial expression. The most well-known methods for face recognition are the eigenface [1-2] and fisherface method[3-5]. The eigenface method is performed by using feature vector transformed by principal component analysis (PCA). The second fisherface method is known as insensitive to large variation in light direction and face pose by using Fisher's linear discriminant (FLD). Here, FLD is a statistical method that maximizes the ratio of between-scatter matrix and within-scatter matrix.

However, the various methods of face recognition including eigenface, fisherface, and computational intelligence usually give equal importance in deciding the face to be recognized, regardless of typicalness. These methods frequently often cause difficulty in those cases where the face images overlap due to illumination and viewing direction. Also, once an input vector is assigned to a class, there is no indication of its strength of membership in that class. Hence, we adopt the fuzzy-based fisherface method, which is an expanded approach of the fisherface method combined the theory of fuzzy sets. Fuzzy sets were introduced by Zadeh [6]. Here, an

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 924–932, 2004.
© Springer-Verlag Berlin Heidelberg 2004

assignment of fuzzy membership value is performed by Fuzzy k-Nearest Neighbor (FKNN) initialization [7].

This paper is organized as follows. Section 2 describes the fuzzy-based Fisher recognition with the wavelet transform. In Section 3, we describe the face recognition experiments for expressive face image and show their results. Section 4 makes some concluding remarks.

2 Fuzzy-Based Fisherface Recognition with Wavelet Transform

Principal Component Analysis (PCA) is a well-known technique in multivariate linear data analysis. While PCA is commonly used to project face pattern from a high-dimensional image space to a lower-dimensional space, a drawback is that is defines a subspace such that it has the greatest variance of the projected sample vectors among all the subspaces. However, such projection may not be effective for classification since large and unwanted variations may be retained. Consequently, the projected samples for each class may not by well clustered, and instead the samples may be smeared together. Linear Discriminant Analysis (LDA) is an example of a class-specific method that finds the optimal projection for classification, which can be briefly described as follows: we consider c class problem with N vectors. Let the between-class scatter matrix be defined as

$$S_B = \sum_{i=1}^{c} N_i (\mathbf{m}_i - \overline{\mathbf{m}})(\mathbf{m}_i - \overline{\mathbf{m}})^T \tag{1}$$

where N_i is the number of vectors in i'th class C_i and $\overline{\mathbf{m}}$ is the mean of all vectors, $\mathbf{m}_i$ is the mean of vectors transformed by PCA in class C_i.

The within-class scatter matrix is defined as follows

$$S_W = \sum_{i=1}^{c} \sum_{x_k \in C_i} (\mathbf{x}_k - \mathbf{m}_i)(\mathbf{x}_k - \mathbf{m}_i)^T = \sum_{i=1}^{c} S_{W_i} \tag{2}$$

where S_{W_i} is the covariance matrix of class C_i. The optimal projection matrix W_{FLD} is chosen as the matrix with orthonormal columns that maximizes the ration of the determinant of the between-class matrix of the projected samples to the determinant of the within-class scatter matrix of the projected sampled, i.e.,

$$W_{FLD} = \arg\max_{W} \frac{\left| W^T S_B W \right|}{\left| W^T S_W W \right|} = [\mathbf{w}_1 \quad \mathbf{w}_2 \quad \cdots \quad \mathbf{w}_m] \tag{3}$$

where $\{\mathbf{w}_i \mid i = 1,2,\cdots,m\}$ is the set of generalized eigenvectors of S_B and S_W corresponding to the $c-1$ largest generalized eigenvalues $\{\lambda_i \mid i = 1,2,\cdots,m\}$, i.e.,

$$S_B \mathbf{w}_i = \lambda_i S_W \mathbf{w}_{i,i=1,2,\ldots,m} \tag{4}$$

Thus, the feature vectors $V = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_N)$ for any face images $\mathbf{z}_i$ can be calculated as follows

$$\mathbf{v}_i = W_{FLD}^T \mathbf{x}_i = W_{FLD}^T E^T (\mathbf{z}_i - \bar{\mathbf{z}}) \tag{5}$$

Now, consider the fuzzy-based LDA method. Let feature vectors transformed by PCA, $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$, be a set of N labeled vectors. The procedure to assign fuzzy membership degree to feature vector transformed by PCA is as follows.

[Step 1] Obtain the Euclidean distance matrix between feature vectors of training sets.

[Step 2] Set diagonal elements to infinite (large value) in the distance matrix because of zero value in $i = j$ case.

[Step 3] Sort the distance matrix in ascending order. And then, select the class corresponding to from 1 to k'th nearest point.

[Step 4] Compute the membership grades for j'th sample point using the following equation.

$$\mu_{ij}(x) = \begin{cases} 0.51 + 0.49(n_{ij}/k), & \text{if } i = \text{the same as the label} \\ 0.49(n_{ij}/k), & \text{if } i \neq \text{the same as the label} \end{cases} \tag{6}$$

Where the value n_{ij} is the number of the neighbors belonging to the i'th class in j'th data. For more detail description of fuzzy-based LDA, refer the paper[8].

Fig. 1(a) and (b) show the misclassification results caused from the happy face image. However, Fig. 1(c) shows better recognition performance due to assigning a fuzzy membership degree to feature vector transformed by PCA.

3 Experiments and Results

Using the wavelet transform, face images can be decomposed into several subband frequency images. Fig. 2 shows an example of a decomposed image into third level. Sub-images LL3, HL3, LH3, and HH3 are third level wavelet transformed results and correspond to LL, HL, LH, and HH frequency bands, respectively[9]. In this work, we used the LL4 band among the fourth level decomposed images.

left : test image, right : recognized image
(a) Eigenface method

left : test image, right : recognized image
(b) Fisherface method

left : test image, right : recognized image
(c) Fuzzy-based fisherface method

Fig. 1. Comparison of face recognition results

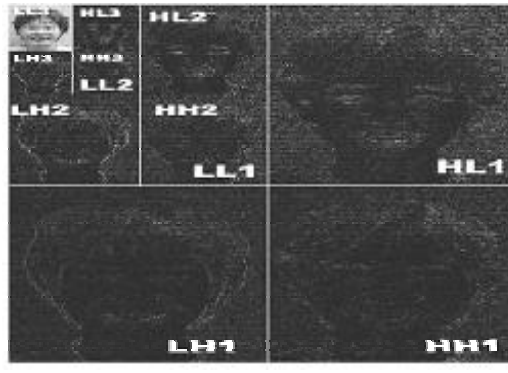


Fig. 2. Subband images from the wavelet transform

After decomposing the face images using the wavelet transform, we first perform a dimensionality reduction by applying PCA. We then search for the most discriminant projection along eigenvectors by successively selecting the principal components. Finally, fuzzy-LDA is performed for this eigenspace to generate a (c-1)-dimensional discriminant subspace. Figure 3 shows the face image expressed by linear combination of eigenfaces.

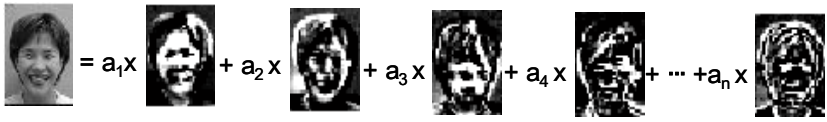


Fig. 3. Eigenface using the PCA

In this work, we have used the Discrete wavelet transform for the training images of 640×480 and acquired the decomposed images of 40×30 from the LL4 band. Thereafter, we applied the PCA method for the LL4 to obtain 91 eigenfaces. And then we adopted the fuzzy-LDA method to get 17 fisherfaces. For the face recognition of expressionless face images, we have acquired 200 expressionless face images from 20 subjects(10 images per subject). Among them, 100 images were used for training and others for testing. These images vary in position and rotation. Here, the changes in scale have been achieved by changing the distance between the person and the video camera. For some individuals, the images were taken at different times, varying facial details. Each image was digitized and presented by a 640×480 pixel array whose gray levels ranged between 0 and 255. Some samples of the expressionless images are shown in Fig. 4. The overall recognition processes used in this work are illustrated in Fig. 5.

Fig. 4. Samples of expressionless face image

Table 1 shows the comparison of recognition rates. For the obtained images, we attempted to reduce the image size by applying the wavelet transform and simply resize by using the resampling with the linear interpolation. We applied several recognition algorithms for the wavelet transformed images (Case A) and resized images (Case B). For the most of the cases, the recognition rates are higher than 95%. Especially, most of the algorithms show perfect recognition results for Case A. From these, we can say that it is difficult to find difference between Fisherface method and

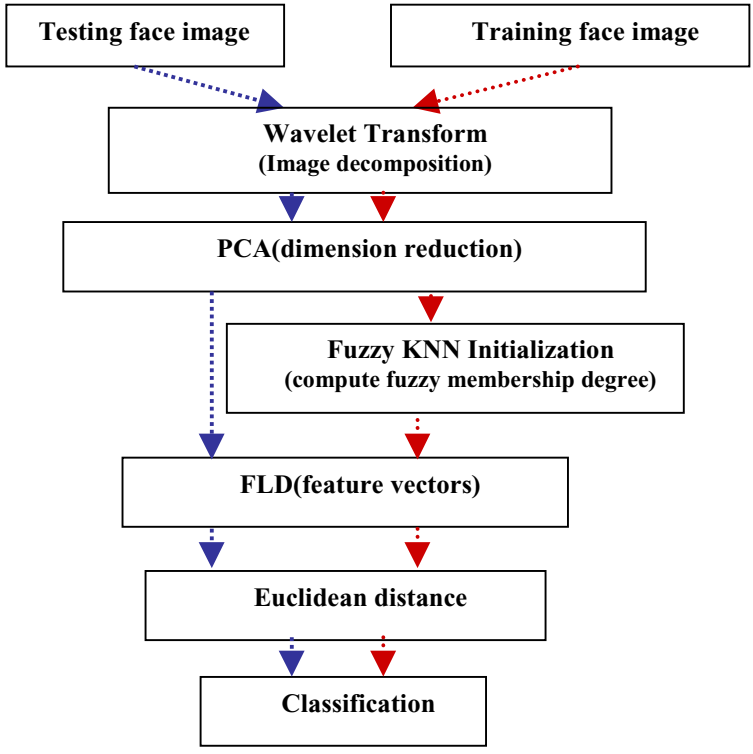


Fig. 5 Flowchart for the fuzzy-based fisherface method

the fuzzy-based fisherface method for expressionless faces. Therefore, for getting more high recognition rate, it is recommended to use the expressionless face images for personal identification.

On the other hand, we also applied the face recognition algorithm for the expressive face images. Capturing the images, we request the subjects to naturally express their emotion. For the case of expressive face images, we captured 5 face images for each emotional state such as happiness, sadness, and anger which were used as testing data (see Fig.6 ~ Fig. 8). For the training images, we used the same images of expressionless case. The expressive face images go through the same preprocessing process as those of expressionless cases.

We denote the experimental results in Table 2. We find some noticeable experimental results. The recognition rates are heavily affected by the facial expressions. Moreover, the recognition rates of anger case become below 85% for all algorithms. However, when we used the fuzzy LDA with wavelet transform, one can get higher recognition rate for expressive face as well as expressionless case.

Table 1. Comparison for recognition rates (Case A: Wavelet transform is applied Case B: No transform is applied)

	Eigenface (PCA)	Fisherface (PCA+LDA)	Fuzzy-based fisherface (PCA+ Fuzzy+ LDA)
Case A	99%	100%	100%
Case B	94%	96%	96%

Table 2. Comparison for recognition rates(Case A: wavelet transformed image, Case B: resized image using a linear interpolation)

		Eigenface (PCA)	Fisherface (PCA+LDA)	Fuzzy-based fisherface (PCA+ Fuzzy+ LDA)
Happy	Case A	87%	94%	96%
	Case B	83%	91%	93%
Sad	Case A	93%	95%	97%
	Case B	91%	92%	94%
Anger	Case A	79%	82%	84%
	Case B	76%	81%	82%

Fig. 6. Samples of happy face images

Fig. 7. Samples of sad face images

Fig. 8. Samples of anger face images

4 Concluding Remarks

In this work, we have studied the face recognition for expressive face images. The previous methods of face recognition including eigenface, fisherface, and computational intelligence give same importance in deciding the face to be recognized regardless of typicalness. However, the fuzzy-based method assigns fuzzy membership to a feature vector transformed by PCA of face image rather than assigning the vector to particular class. Thus, we can reduce the sensitivity between the face images due to expression.

Today, we live in a very rapid changing information society. Under the information society, a lot of beneficial information services are provided through the interconnected network. In this situation, unauthorized user often destructs the information system and unveils the privacy and spread unsound information. To tackle this problem, the robust face recognition algorithm as a biometrics technique is required now. Here, we should consider the expressive face images which is often captured in real situations. We hope this study will be a good starting point of establishing more practical face recognition system considering expressive faces.

Acknowledgements. This work was supported by grant No. R01-2002-000-00315-0 from the Basic Research Program of the Korea Science & Engineering Foundation.

References

1. M. Turk, A. Pentland, Face recognition using eigenfaces, *Proc. IEEE Conf. On Computer Vision and Pattern Recognition*, 1991, 586-591.
2. M. Turk, A. Pentland, Eigenfaces for recognition*, Journal of Cognitive Neuroscience*, Vol. 3, No. 1, pp. 71-86, 1991.
3. P. N. Belhumeur, J. P. Hespanha, D. J. Kriegman, Eigenfaces vs. Fisherfaces: recognition using class specific Linear Projection, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7), 1997, 711-720.
4. P. Belhumeur, J. Hespanha, D. Kriegman, Eigenfaces vs. fisherfaces: Recognition using class specific linear projection, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 711-720, 1997.
5. C. Padgett, G. Cottrell, Representing face images for emotion classification, *Advances in Neural Information Processing Systems*, Vol. 9, MIT Press. 1997.
6. L. A. Zadeh, Fuzzy sets, *Information and Control*, 8, 1965, 338-353.
7. J. M. Keller, M. R. Gray, J. A. Givens, A fuzzy k-nearest neighbor algorithm, *IEEE Trans. on Systems, Man, and Cybernetics*, 15(4), 1985, 580-585.
8. Keun-Chang Kwak and Witold Pedrycz, Face Recognition Using Fisherface Classifier, *Submitted to IEEE Trans. on Fuzzy systems*.
9. Hyoun-Joo Go, Keun-Chang Kwak, Dae-Jong Lee, Myung-Geun Chun, Emotion Recognition From the Facial Image and Speech Signal, *SICE Annual Conference in Fukui, 2003*

Kolmogorov-Smirnov Test for Image Comparison

Eugene Demidenko

Dartmouth College, Hanover, NH 03755, USA
eugened@dartmouth.edu
http://www.dartmouth.edu/~eugened

Abstract. We apply the Kolmogorov-Smirnov test to test whether two distributions of 256 gray intensities are the same. Thus, this test may be useful to compare unstructured images, such as microscopic images in medicine. Usually, histogram is used to show the distribution of gray level intensities. We argue that cumulative distribution function (gray distribution) may be more informative when comparing several gray images. The Kolmogorov-Smirnov test is illustrated by hystology images from untreated and treated breast cancer tumors. The test is generalized to ensembles of gray images. Limitations of the Kolmogorov-Smirnov test are discussed.

1 Introduction

An essential task of image analysis is image comparison. Surprisingly, no statistical tests are available to compare images with a fixed type I probability error (the error to reject the true null hypothesis). Under the assumption that images are subject to random noise, we want to test if two images have the same grayscale distribution. Clearly, if two images are the same, up to a small noise, they have close grayscale distributions. The reverse is not true. Thus, grayscale distribution analysis is helpful when images of the same content are compared. This test is especially useful to compare microscopic images of tissue or other *unstructured* images frequently used in biology in medicine.

The image histogram is a frequently used technique of image processing [2]. However, besides the histogram one can compute the distribution, or more specifically the cumulative distribution function as the sum of the probabilities that a pixel takes the grayscale level less than g, where $g = 0, ..., 255$. In fact, when statistical analysis is concerned with the distribution of a random variable the empirical cumulative distribution function is usually used, not the histogram as an estimate of the density [8]. One explanation is that while the estimation of the distribution function is unbiased and straightforward the estimation of the density is not, and moreover leads to an ill-posed problem; see [10] for detail.

Let $\{h_g, g = 0, 1, ..., 255\}$ be the histogram of a gray image as the probability that the gray level takes the intensity value $g = 0, 1, ..., 255$. The empirical cumulative gray level distribution function, or shortly *gray distribution* (gd), is defined as follows

$$F_g = \sum_{g'=0}^{g} h_{g'}.$$

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 933–939, 2004.
© Springer-Verlag Berlin Heidelberg 2004

Function F_g is a nondecreasing step-function with the step $1/256$ at $g = 0, 1, ..., 255$, see the right panel of Figure 2 as a typical gray distribution. An advantage of the gd analysis is that it facilitates visual image comparison by plotting grayscale distribution functions on the same scale. Indeed, it is difficult to plot several histograms on the same scale because they often overlay each other–see the left panel of Figure 2 for an example. However, besides better visualization an advantage is application of nonparametric statistical tests, such as Kolmogorov-Smirnov test. Other nonparametric tests for distribution comparison are available, such as Friedman or Wilcoxon test, as described in the reference book [3]. However, the Kolmogorov-Smirnov test is the most popular, perhaps due to simplicity of computations.

Several authors used Kolmogorov-Smirnov test for image segmentation. See [5], [7], [6], to name a few. We apply this nonparametric test to statistical image comparison, or more precisely to test whether images have the same gray distribution.

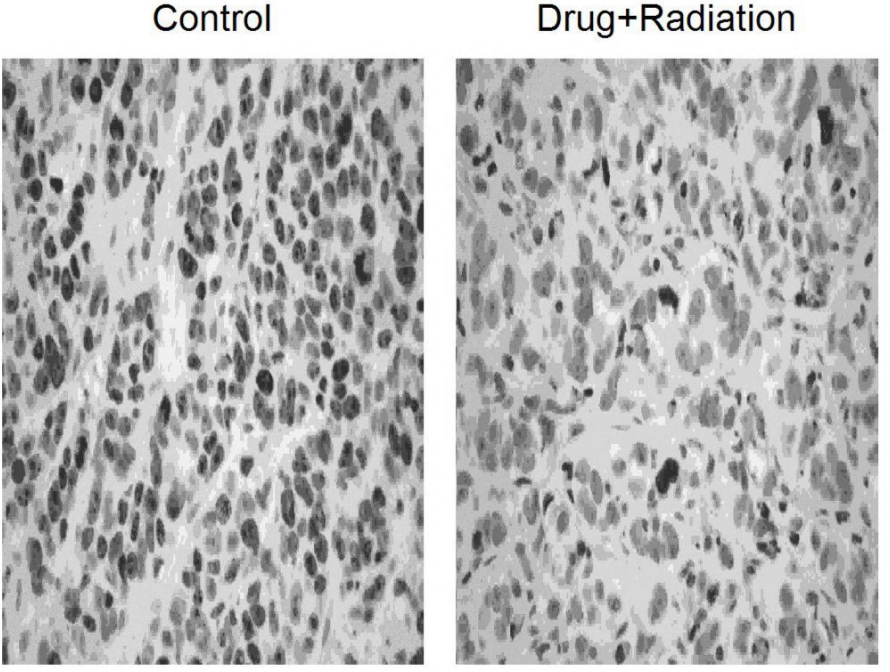


Fig. 1. Histology sections of untreated (control group) and treated tumors. The living cancer cells are the dark spots (blobs). To statistically test that the two images have the same gray distributions the Kolmogorov-Smirnov test is applied.

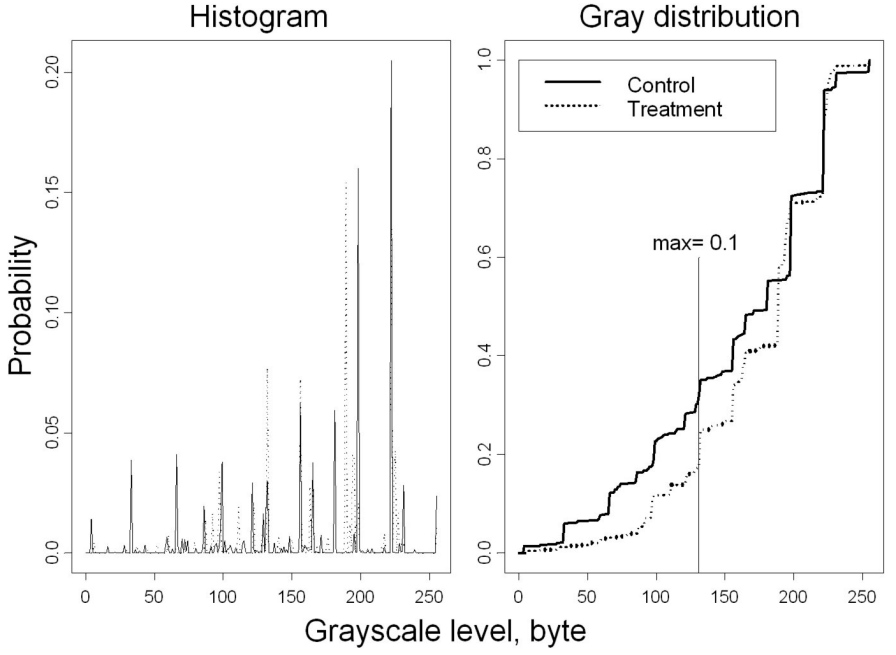


Fig. 2. Histogram and gray distributions for two histology images. The distribution function of the treated tumor is less (almost everywhere) than control (maximum difference is 0.1) which means that the control image is darker. The Kolmogorov-Smirnov distance is used to test the statistical significance.

2 Kolmogorov-Smirnov Test for Image Comparison

Let $F^{(1)} = \{F_g^{(1)}, g = 0, ..., 255\}$ and $F^{(2)} = \{F_g^{(2)}, g = 0, ..., 255\}$ be two gray distributions for $P_1 \times Q_1$ and $P_2 \times Q_2$ images M_1 and M_2. We compute the maximum,

$$\widehat{D} = \max_g \left| F_g^{(1)} - F_g^{(2)} \right|,$$

the distance of one empirical distribution from the other. Kolmogorov [4] and Smirnov [9] proved that the probability that $\widehat{D} > D$, i.e. the observed distance is greater than the threshold, is

$$Q_{KS}(\lambda) = 2 \sum_{j=1}^{\infty} (-1)^{j-1} \exp\left(-2j^2\lambda^2\right),$$

where $\lambda_{KS} = D[\sqrt{J} + 0.11/\sqrt{J} + 0.12]$ and

$$J = \frac{P_1 Q_1 P_2 Q_2}{P_1 Q_1 + P_2 Q_2}.$$

Thus, $Q_{KS}(\lambda)$ with D replaced with $\widehat{D}$ may be treated as the p-value of the test. We notice, the greater the distance between the two distributions the greater the value of $\widehat{\lambda}_{KS}$ and lesser is the probability $Q_{KS}(\widehat{\lambda}_{KS})$. For example, if two images yield distance $\widehat{D}$ and the computed probability $Q_{KS}(\widehat{\lambda}_{KS}) < .05$, we reject the hypothesis that the two images are the same with a 5% error. We can find λ_{KS} such that $Q_{KS}(\lambda_{KS}) = 0.05$ which yields the threshold $\lambda_{KS} = 1.358$.

As a word of caution, all nonparametric tests, including Kolmogorov-Smirnov, have the alternative $H_A : F_1(x) \neq F_2(x)$ for at least one x. Therefore, this test may be conservative.

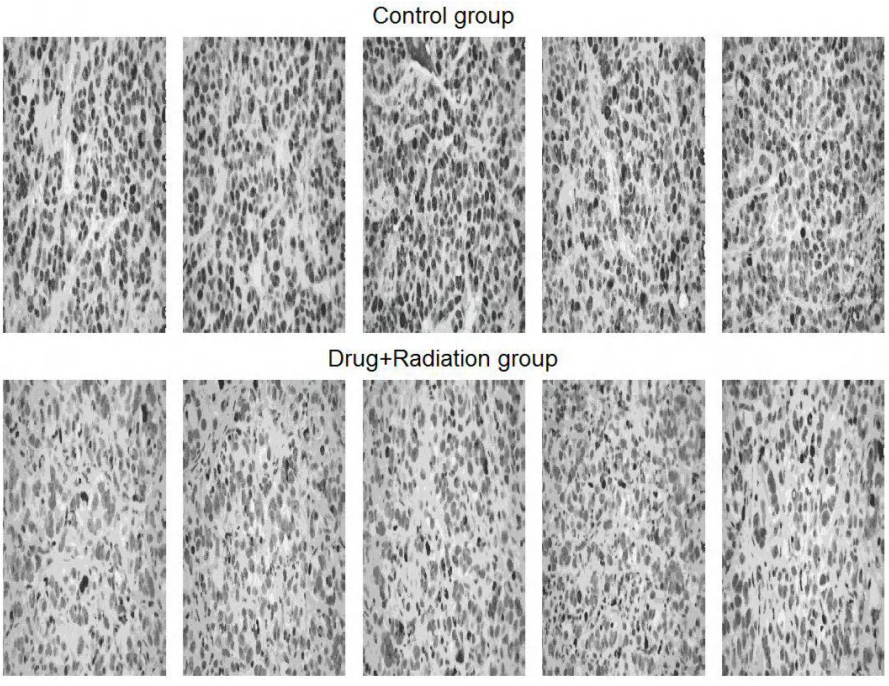


Fig. 3. Two ensembles of histology images taken from 10 mouse tumors in two groups. To determine the cancer kill effect these images are compared by the Kolmogorov-Smirnov test. The darker the image more living cancer cells.

3 Example: Histological Analysis of Cancer Treatment

We illustrate the Kolmogorov-Smirnov test by a histological analysis of breast cancer treatment as described in [11]. Two 2048×1536 images of proliferative activity tumor tissue sections are shown in Figure 1. Dark blobs are cancer cells. In the control tumor (left panel) no treatment was given. In the treated tumor

(right panel) a combination of drug, EB 1089 plus radiation, seems to reduce the number of living cancer cells. We want to confirm this reduction statistically using the Kolmogorov-Smirnov test by computing the p-value.

The grayscale histogram and the distribution functions for these images are shown in Figure 2. Clearly, it is difficult to judge the difference in the images by histogram. To the contrary, the distribution functions reveal the difference with the absolute maximum 1/10 at gray level $g = 131$. We notice that the treatment distribution function is below control (for most gray levels) which means that the right image is lighter. For these images $P_1Q_1 = P_2Q_2 = 2048 \times 1536 = 3.1457 \times 10^6$, yielding $\widehat{\lambda}_{KS} = 186.96$ and $Q(\widehat{\lambda}_{KS}) < 0.0001$, near zero. Since the p-value is very small, we infer that the null hypothesis that two images are the same should be rejected. This means that the kill effect of combination of drug and radiation is significant.

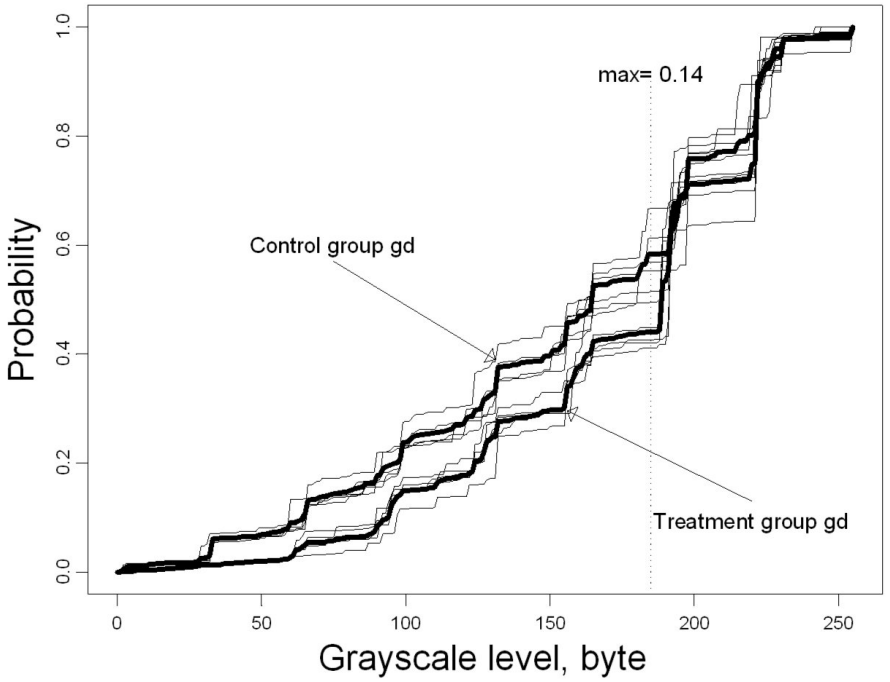


Fig. 4. Gray distributions for 10 histology images in two groups. The bold line shows the groups mean gd. The distance 0.14 between group gds is attained at $g = 185$.

4 Ensemble of Images

In many instances we deal with repeated images or ensemble of images. For example, to determine the kill effect histology images may be taken at different

tumor sites, from different animals, etc. In Figure 3, we show histology from 10 different animals, 5 in each group. While human eye may detect difference between two images it becomes difficult to judge several images. Then multivariate statistical testing becomes essential. If two ensembles of images are compared, we may assume that the members within one ensemble have identical distribution. Then each ensemble can be treated as a pooled sample of gray levels. If images are of the same size, it is elementary to show that the group gd is the arithmetic mean of individual gds. Therefore, the group comparison reduces to a comparison of two gray distribution.

In Figure 4 we show 10 individual gds and two group gds (bold). The difference between two gds is 0.14 and is attained at $g = 185$. The p-value of the Kolmogorov-Smirnov test is less than 0.0001. This confirms that the treatment kills statistically significant number of cancer cells.

5 Discussion

We have demonstrated that the Kolmogorov-Smirnov test can be used for image comparison. This test may be used to compare two images or ensemble of images with the null hypothesis that two images (or two samples of images) have the same distribution of 256 gray level intensities. This test is useful to compare content-free or unstructured images when human eye is not able to detect the difference. In particular, the Kolmogorov-Smirnov test can be applied to compare treatment groups in biology and medicine when several dozens or even hundreds of microscopic images, such as histology images, are compared. Showing several gray distributions on one plot is more feasible than histograms and as such it is more convenient for visual comparison.

It is worthwhile to remember that the Kolmogorov-Smirnov test is a two-sided test so we cannot test the hypothesis that one image is darker than another. Also, the Kolmogorov-Smirnov test is a stringent test because the hypothesis is rejected if at least one out of 256 gray level intensities is different.

A limitation of the Kolmogorov-Smirnov test applied to two ensembles of images is the assumption that members from one groups have identical gray distributions and there is no room for site or animal heterogeneity. To account for individual image variation more advanced methods of mixed models should be employed. The interested reader is referred to a recent book [1], where this methodology is applied to image analysis and comparison.

References

1. Demidenko, E. (2004). *Mixed Models: Theory and Applications*. New York: Wiley.
2. Gonzalez, R.C. and Woods, R.E. (2002). *Digital Image Processing*. Second edition. Upper Saddle River, NJ: Prentice Hall.
3. Hollander M. and Wolfe, D.A. (1999). *Nonparametric Statistical Methods*. New York: Wiley.
4. Kolmogorov, A.N. (1941). Confidence limits for an unknown distribution function. *Annals of Mathematical Statistics* **12**, 461-483.

5. Koster, K. and Spann, M. (2000). MIR: An approach to robust clustering - Application to range image segmentation. *IEEE Transactions in Pattern Analysis* **22**: 430-444.
6. Lozano, M.A. and Escolano, F. (2003). Two new scale-adapted texture descriptors for image segmentation. *Lecture Notes in Computer Science* 2905: 137-144.
7. Pauwels, E.J. and Frederix, G. (2000). Image segmentation by nonparametric clustering based on the Kolmogorov-Smirnov distance. *CiteSeer*: http://citeseer.nj.nec.com/pauwels00image.html.
8. Sheskin, D.J. (2004). *Handbook of Parametric and Nonparametric Statistical Procedures*. Boca Raton: Chapman and Hall.
9. Smirnov, N.V. (1948). Table for estimating the goodness of fit of empirical distribution. *Annals of Mathematical Statistics* **19**, 279-281.
10. Silverman, B. W. (1986). *Density Estimation for Statistics and Data Analysis*. London: Chapman and Hall.
11. Sundaram, S., Sea, A., Feldman, S., Strawbridge, R., Hoopes, P.J., Demidenko, E., Binderup, L., and Gewirtz, D.A. (2003). The combination of a potent vitamin D_3 analog, EB 1089, with ionizing radiation reduces tumor growth and induces apoptosis of MCF-7 breast tumor xenografts in nude mice. *Clinical Cancer Research* **9**, 2350-2356.

Modified Radius-Vector Function for Shape Contour Description

Sung Kwan Kang[1], Muhammad Bilal Ahmad[2], Jong Hun Chun[3], Pan Koo Kim[4], and Jong An Park[1]

[1] Dept. of Information & Communications Engineering,
Chosun University, Gwangju, Korea.
japark@chosun.ac.kr

[2] Signal and Image Processing Lab, Dept. of Mechatronics,
Kwangju Institute of Science and Technology, Gwangju, Korea
bilal@kjist.ac.kr

[3] Provincial College of Namdo, Korea.

[4] College of Electronics and Information Engineering,
Chosun University, Gwangju, Korea.

Abstract. Shape is one of the salient features of visual content and can be used in visual information retrieval. Radius vector function for star-shaped objects is available. A modified radius-vector function for both types of shapes-star shaped, and non-star shaped is presented. The center of gravity is selected as the reference point. The corner points are marked as the nodes. Normalized vectors are calculated from the reference point to the corner nodes. The normalized vectors are arranged in ascending order, and the Euclidean distance of the query object with database objects is used for shape matching criteria.

1 Introduction

Pattern recognition is the area that studies the operation and design of systems that recognize patterns in data [1]. Important application areas include image analysis, character recognition, speech analysis, man and machine diagnostics, person identification and industrial inspection. In recent years shape recognition has received a lot of interest. It plays an important role in a wide range of applications. For example, imagine that you have a large number of images and wish to select some of them, which are similar to a certain image. By using some image properties such as color, texture and shape structure, we can retrieve all images that are alike or so.
Shape is one of the salient features of visual content and can be used in visual information retrieval [2]. Shape analysis is useful in a number of applications of machine vision, including medical image analysis, aerial image analysis, and manufacturing. Object recognition techniques can be classified as local and global techniques. Global techniques are based on the use of global features of objects such as Fourier descriptors [3] and Moments [4]. Local techniques use local features such as critical points. These local techniques are especially useful in the presence of noise or in the case of partial occlusion. Other criteria to discriminate between object

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 940–947, 2004.
© Springer-Verlag Berlin Heidelberg 2004

recognition techniques are the region-based and the boundary-based techniques (also referred to as internal and external techniques [5]-[6]). Region-based techniques deal with the region in the image that corresponds to the object under consideration. Boundary-based techniques trace the boundary only, while ignoring the interior of the object.

Shape description and its corresponding matching algorithm is one of the main concerns in MPEG-7. To fully exploit the possibilities of MPEG-7 descriptions, automatic extraction of features (or 'descriptors') will be extremely useful. MPEG-7 visual description tools consist of basic structures and descriptors that cover the basic visual features such as color, texture, shape, motion, localization, etc. In order to retrieve an image from a large database the descriptor should have enough discriminating power and immunity to noise. In addition, the descriptor should be invariant to scale, translation, rotation and reflection [7]-[10]. After the objects with closed contours have been detected, a corresponding mechanism must be established to match these objects. One of the useful shape descriptors is based on the radius vector function.

In this paper we will recognize objects by shape matching using modified radius-vector functions. Radius vector function for star-shaped objects is available in the literature [11], which can only be used for star-shaped objects. A modified radius-vector function can be used for both types of shapes-star shaped, and non-star shaped. The center of gravity is selected as the reference point. The corner points are marked as the nodes. Normalized vectors are stored from the reference point to the corner nodes. The normalized vectors are rearranged in ascending order. The Euclidean distance of the center to the corner distances of test object with those of objects in the database is used for shape matching.

This paper is organized as follows. Section 2 describes briefly about the traditional radius vector function, and explains the proposed algorithm for modified radius vector function. The simulation results are shown in section 3.

2 The Proposed Algorithm

Before defining the modified radius vector function, we found the appropriate global reference point and reference line. The center of gravity and the principal axes could be the best reference point and line, respectively [3]. The center of gravity and the principal axes are found with the help of statistical moments as below:

The moments of a binary image b(x, y) are given by

$$\mu_{pq} = \sum_{x} \sum_{y} b(x, y) x^{p} y^{q}, \tag{1}$$

where p and q define the order of moment. Where b(x,y) can be omitted in the case of binary images, so sums are only taken where b(x,y) has values 1, and the equation reduces to

$$\mu_{pq} = \sum_{x} \sum_{y} x^{p} y^{q}, \tag{2}$$

The center of gravity of the object can be found from moments as:

$$\bar{x} = \frac{\mu_{10}}{\mu_{00}}, \qquad \bar{y} = \frac{\mu_{01}}{\mu_{00}}, \tag{3}$$

where $(\bar{x}, \bar{y})$ are the coordinates of the center of gravity. The pqth discrete central moment m_{pq} of a region is defined by

$$m_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q , \tag{4}$$

where the sums are taken over all points (x,y). Zero-order moment m_{00} represents the binary object area. Second order moments express the distribution of matter around the center of gravity, and are called moments of inertia.

The contour of a shape is described by the radius-vector function defined in the following way. A reference point *O* in the interior of the figure 1 is selected. The radius vector function R(θ) is then the distance from the reference point *O* to the contour in the direction of the θ – ray, where 0 θ 2π.

It is, however, necessary that the figure is star-shaped with respect to *O*; that is, for any contour point *P* the whole line segment from *O* to *P* lies within the figure. In this case the radius-vector function completely characterizes the shape: if R(θ) is given then the shape can be completely reconstructed. If the star-shaped-ness is violated only by small irregularities in the contour it is possible to recover it by smoothing. In the general case, however, the description by the radius-vector function is not suitable for non-star-shaped objects.

Before explaining the proposed algorithm, we need to find the corner points of the object. The corner points of an object can be found by rotating a similar radius vector R(θ), as shown in figure 1. The distances of the boundary points from the center of gravity are found by rotating the radius vector from 0 to 360 degree. A plot of distance verses angle is drawn. The corner points are found from the maximum and minimum points from the distance-angle graph. When the radius vector is moving, the distance is either decreasing or increasing. At the corner points, the distance changed its direction, and that is the corner point. Figure 2 shows such a graph for an object. The corner points can be easily found from the maximum and minimum points.

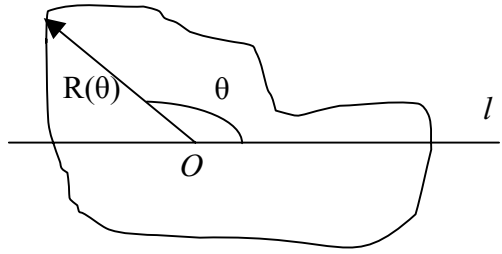


Fig. 1. Radius-vector FUNCTION

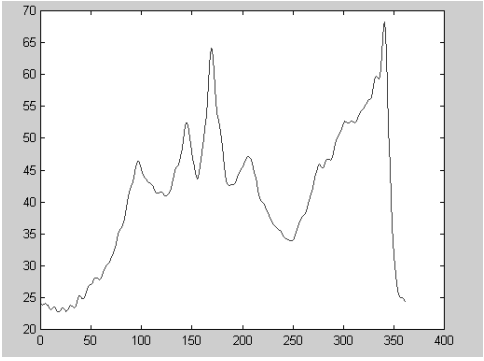


Fig. 2. A distance-angle plot for an object to find corner points

If there is smooth transition at corner due to noise or blur, the algorithm defined in figure 2 detects the corners. The procedure defined in figure 2 detects corners when there is a change of increasing or decreasing trend in the distance-angle plot. Therefore the procedure defined in figure 2 efficiently detects the smooth corners, as one can see that the maximum or minimum points in the figure 2 or not very sharp at every extreme locations.

In this paper, we proposed a modified radius-vector function for both types of shapes-star shaped, and non-star shaped. The center of gravity using (3) is selected as the reference point *O*. The corner points are marked as the nodes. The radius vector function is defined as R(N), where 0 N number of nodes. For the shape description to be invariant under the scaling, translation, rotation and reflection transformations, we store the descriptor distances, as follow.

Figure 3 shows the shape of object with nodes marked from 1 to N, and the corresponding vectors from center to node are marked from l_1 to l_N. The average center to node distance is calculated as:

$$L_A = \frac{1}{N}\sum_{i=1}^{N} l_i \tag{5}$$

For scaling transformation invariant, each center to node vector is normalized with the average distance as:

$$\left\langle l_{A,i} \right\rangle = l_i \Big/ L_A \tag{6}$$

To get rotation invariant, the normalized vectors are sorted in the ascending (or descending) order and are placed together with highest distance at the right side. This will disturb the original sequence of the normalized vectors. This disturbance will give the rotation invariant property. Two images of a same object with different rotations will have different sequence of the vectors. For example, one image might have the highest value from the reference point to node N3 whereas in other view of the same object it might have the highest value from the reference point to node N6

because of rotation. With sorting, both images will have the vectors of highest value at the same position. Similarly, all the vectors will be placed in ascending order. The sorted vectors of the modified radius vector function are scale, rotation and reflection invariants.

If some corners are missing or finding more, then the algorithm could be failed. But due to sorting of the vectors, the position of the missing/or finding corners in the sorted vectors will depend on their vector distance. If the distance is small, the algorithm will not be effected, but if its distance is large than the algorithm might failed. Similarly, the sorting of vectors disturb the actual shape of the object, and could mislead the recognition, but its necessary for the rotation, reflection, noise robustness attacks. This algorithm will result in both correct and incorrect results, and the feedback from the user would be helpful for efficient retrieval.

To find the difference (or distance) of two modified radius vector function, we used the Euclidean distance measure. There are many other distance measures such as sum of absolute distance, but the Euclidean distance is very popular. The Euclidean distance measure is defined as the square root of the sum of the squared differences between two feature vectors, one belonging to the query's shape, Q, and the other belonging to a shape in the contents' database, C, for which a description is available.

$$D_E = \sqrt{\sum_{k=1}^{N} (Q[k] - C[k])^2}, \tag{7}$$

where Q[k](k = 1, 2, 3, …,N) and C[k] (k = 1, 2, 3, …,N) denote the sorted normalized distances for the query image Q and the database image C, respectively. N represents the number of features in the feature vector. Here N corresponds to the number of vectors or nodes in the image.

Euclidean distance is used to account for the small noise or occlusion presents in the images. If there is occlusion, some part of the boundary of the image will might not be visible, and position of corner points will be changed. Also in the presence of noise, the corners position might be slightly displaced. The change of corner positions is related with the amount of occlusion, and noise. Because of occlusion and noise, some of the vectors will have different values than that of the actual object in the database. The Euclidean distance will give a very small value depending on the amount of occlusion and noise. If there is no occlusion and noise, the Euclidean distance will be equal to zero. The minimum Euclidean distances are used as matching criteria to account for the noise and occlusion problems.

For matching the shape descriptor, the normalized center to node vectors of each shape are already stored in the database, i.;e; $\langle l_{D,i} \rangle$, where 0 D No. of shapes in the database. For decision, the Euclidean distance is calculated between the query and the database shapes using (7). The matched object is decided as:

$$Matched = Object(Min(Euclidean(D, A)) \tag{8}$$

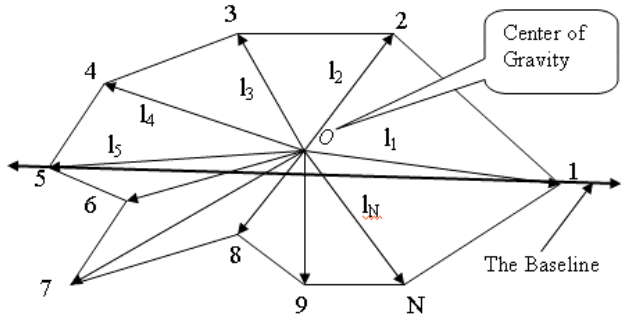


Fig. 3. Shape A marked with nodes and distances

3 Simulations

The proposed algorithm was tested on various 2D synthetic star-shaped and non-star shaped shapes. The simulation results show high matching rate for both types of shapes. Figure 4 shows a synthetic image. The vectors from the center of gravity to the corner points are shown in figure 4. The normalized vectors of figure 4 are shown in figure 5 (a). The horizontal axis of figure 5 shows the node number and the vertical axis shows its normalized distance using (6). The normalized vectors are sorted in ascending order and are shown in figure 5 (b). The numbering of nodes is rearranged to get the rotation invariant property.

The algorithm is also tested on a non-star shaped, as shown in figure 6. Figure 6 shows three different views of a same object. The object is rotated and resized in figure 6 (d) and (g). The corner points are marked with stars and the vectors from the center of gravity to the corner points are marked with straight lines. Figure 6 (b), (e) and (h) show the graph of distances of normalized vectors against the corresponding nodes of objects in figure 6 (a), (d) and (g), respectively. The sorted normalized vectors are shown in figure 6 (c), (f), and (i). The Euclidean distances are measured for the sorted normalized radius vectors. The minimum Euclidean distances are considered as the matched objects. Table 1 shows the Euclidean distance of the test objects as shown in figure 4, and in figure 5. The table shows that the Euclidean distances are very small for similar objects and higher for different objects.

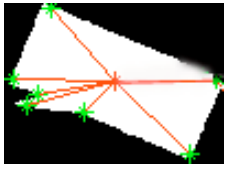
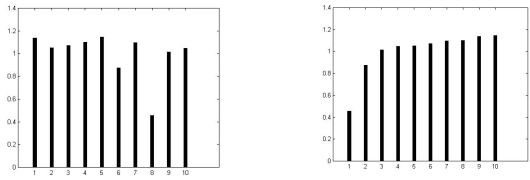
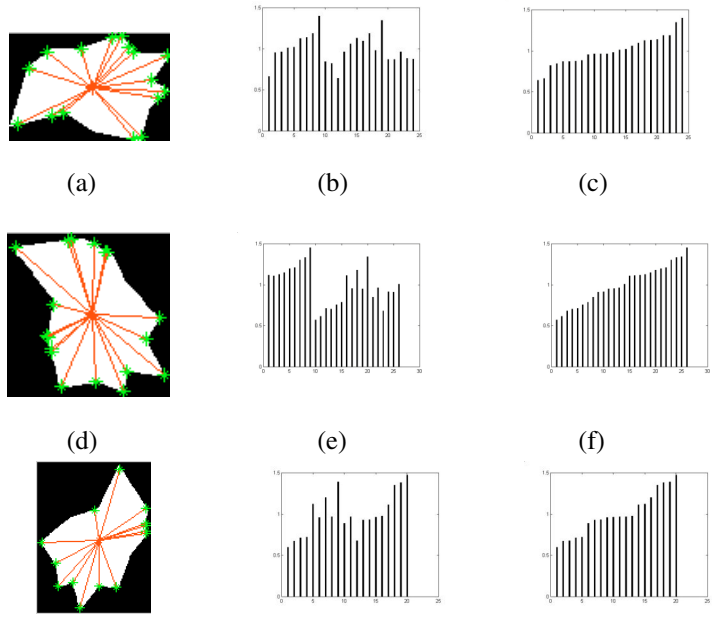


Fig. 4. The vectors from the center of gravity to the corner points for an object.



(a) (b)

Fig. 5. (a) The normalized vectors from the center of gravity to the corner points for figure 4 (b) the sorted normalized vectors.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 6. (a), (d), (g) three different views (rotation and scaling transformations) of an object and their corresponding (b), (e), (h) normalized and (c), (f), (i) sorted radius vectors.

Table 1. Euclidean distance between query object and the database objects for image retrieval of images in figure 4 and 5.

	Fig. 4	Fig. 6 (a)	Fig. 6 (d)	Fig. 6 (g)
Fig. 4	0.0	0.47	0.51	0.49
Fig. 6 (a)	0.47	0.0	0.12	0.09
Fig. 6 (d)	0.51	0.12	0.0	0.13
Fig. 6 (g)	0.49	0.09	0.13	0.0

4 Conclusions

Shape is one of the salient features of visual content and can be used in visual information retrieval. After the objects with closed contours have been detected, a corresponding mechanism must be established to match these objects. One of the useful shape descriptors is based on the radius vector function. Radius vector function for star-shaped objects is available. The proposed modified radius-vector function works very well for both star and non-star shaped. The normalized vectors from the center of gravity to the corner points are used as a measure of shape similarity. The Euclidean distances are calculated for the sorted normalized radius vectors, and the minimum Euclidean distances are used as the criteria for matching objects.

References

1. Morton Nadler, and Eric P. Smith, "Pattern Recognition Engineering" A Wiley Interscience Publication.
2. A. Goshtasby, "Registration of images with geometric distortions", IEEE Trans. Geosci. Remote Sensing, vol. 26, pp. 60-64, Jan. 1988.
3. K. Arbter, W.E. Synder, H. Burkhardt, and G. Hirzinger, "Application of Affine-Invariant Fourier Descriptors to Recognition of 3-D Objects," IEEE Trans, Pattern Analysis and Machine Intelligence, vol. 12, no. 7, pp. 640-647, July 1990.
4. L. G. Brown, "A survey of image registration techniques", Computer Survey, vol. 24, no. 4, pp. 325-376, 1992
5. H. Freeman and J. M. Glass, "Computer Processing of Line Drawing Images", Computing Surveys, 6, No. 1, 57-97, 1974
6. H. Freeman and J. M. Glass, "Boundary Encoding and Processing", in Picture Processing and Psychopictorics, B. S. Lipkin and A. rosenfeld, eds., Academic Press, new York, pp. 241-263, 1970.
7. T. Pavlidis, "Algoritms for Shape Analysis of Contours and Waveforms," IEEE Trans, Pattern Analysis and Machine Intelligence, vol. 2, no. 4, pp. 301-312, April 1980.
8. E. Bribiesca and A. Guzman, "Shape Description and Shape Similarity for Two-Dimensional Regions", in: Proceedings of the Fourth International Join Conference on Pattern Recognition, Kyoto, November 7-10, pp. 608-612, IEEE, 1978.
9. E. Bribiesca and A. Guzman, "How to Describe Pure Form and How to Measure Differences in Shapes Using Shape Numbers", Pattern Recognition, 12, No. 2, 101-112, 1980.
10. Berthold Klaus Paul Horn "Robot Vision" The MIT Electrical Engineering and Computer Science Series.
11. R.C. Gonzalez and R. E. Woods, Digital Image Processing, Addison-Wesley, 1993Allen.

Image Corner Detection Using Radon Transform

Seung Jin Park[1], Muhammad Bilal Ahmad[2], Rhee Seung-Hak [3], Seung Jo Han[3] and Jong An Park[3]

[1] Dept. of Biomedical Engineering,
Chonnam National University Hospital, Gwangju, Korea.
sjinpark@cnuh.chonnam.ac.kr
[2] Signal and Image Processing Lab, Dept. of Mechatronics,
Kwangju Institute of Science and Technology, Gwangju, Korea.
bilal@kjist.ac.kr
[3] Dept. of Information & Communications Engineering,
Chosun University, Gwangju, Korea.
japark@chosun.ac.kr

Abstract. This paper describes a new corner detection algorithm based on the Radon Transform. The basic idea is to find the straight lines in the images and then search for their intersections, which are the corner points of the objects in the images. The Radon Transform is used for detecting the straight lines and the inverse Radon Transform is used for locating the intersection points among the straight lines, and hence determine the corner points. The algorithm was tested on various test images, and the results are compared with well-known algorithms.

1 Introduction

Corners have been found to be very important in human perception of shapes and have been used extensively for shape description, recognition, and data compression [1]. Corner detection is an important aspect of image processing and finds many practical applications. Applications include motion tracking, object recognition, and stereo matching. Corner detection should satisfy a number of important criteria. It should detect all the true corners, and the corner points should be well localized, and should be robust with respect to noise, and should be efficient. Further, it should not detect false corners.

There is an abundance of literature on corner detection. Moravec [2] observed that the difference between the adjacent pixels of an edge or a uniform part of the image is small, but at the corner, the difference is significantly high in all directions. Beaudet [3] proposed a determinant (DET) operator which has significant values only near corners. Kitchen and Rosenfeld [4] presented a few corner-detection methods. The work included methods based on gradient magnitude of gradient direction, change of direction along edge, angle between most similar neighbors, and turning of the fitted surface. Lai and Wu [5] considered edge-corner detection for defective images. Wu and Rosenfeld [6] proposed a technique which examines the slope discontinuities of the x and y projections of an image to find the possible corner candidates. Paler et al.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 948–955, 2004.
© Springer-Verlag Berlin Heidelberg 2004

[7] proposed a technique based on features extracted from the local distribution of gray-level values. Arrebola et al. [8] introduced corner detection by local histograms of contour chain code. Kohlmann [9] applied the 2D Hilbert transform to corner detection. Smith and Brady [10] used a circular mask for corner detection. No derivatives were used. Mehrotra et al. [11] proposed two algorithms for edge and corner detection. The first is based on the first-directional derivative of the Gaussian, and the second is based on the second-directional derivative of the Gaussian. Davies [12] applied the generalized Hough transform to corner detection. Mokhtarian [13] used the curvature-scale-space (CSS) [14] technique to search the corner points. The CSS technique is adopted by MPEG-7. The Kitchen and Rosenfeld detector [4], the SUSAN detector [10] and the CSS [13] corner detector have shown good performance. These detectors are therefore chosen as our test detectors.

In this paper, a new corner detection based on the forward and inverse Radon Transform [15-16] is presented. The straight lines in the images are detected and their intersection points are used to locate the corner points. This paper is organized as follows. Section 2 describes the method to determine the straight lines in the images using Radon Transform and section 3 describes the algorithm to detect the intersection points among the straight lines using the inverse Radon Transform. Section 4 describes the simulation results. At the end, we will conclude our paper with few final remarks.

2 Detecting Straight Lines Using Radon Transform

Radon transform can be efficiently used to search the straight lines in the images. It transforms two dimensional images with lines into a domain of possible line parameters, where each line in the image will give a peak positioned at the corresponding line parameters. The Radon transformation shows the relationship between the 2-D object and the projections. Let us consider a coordinate system shown in Fig. 1. The function $g(s,\theta)$ is a projection of *f(x,y)* on the axis *s* of θ direction. The function $g(s,\theta)$ is obtained by the integration along the line whose normal vector is in θ direction. The value $g(0,\theta)$ is defined that it is obtained by the integration along the line passing the origin of (x,y)-coordinate. The general Radon transformation is given as:

$$g(s,\theta) = \int_{-\infty}^{\infty} f(x,y)\delta(x\cos\theta + y\sin\theta - \rho)dxdy \tag{1}$$

The Eq. (1) is called Radon transformation from the 2-D distribution f(x,y) to the projection $g(s,\theta)$.

Although the Radon transformation expresses the projection by the 2-D integral on the x,y-coordinate, the projection is more naturally expressed by an integral of one variable since it is a line integral. Since, the s,u-coordinate along the direction of projection is obtained by rotating the x,y-coordinate by θ, the Radon transform, after a small change of axes transformation, is given as:

$$g(s,\theta) = \int_{-\infty}^{\infty} f(s\cos\theta - u\sin\theta, s\sin\theta + u\cos\theta)\delta(0)dsdu \tag{2}$$

Since the δ-function in Eq. (2) is a function of variable s, we get

$$\int_{-\infty}^{\infty} \delta(0)ds = 1$$

It follows from the above that the Radon transformation $g(s,\theta)$ in Eq. (1) is translated into the following integral of one variable u,

$$g(s,\theta) = \int \int_{-\infty}^{\infty} f(s\cos\theta - u\sin\theta, s\sin\theta + u\cos\theta)du \tag{3}$$

This equation expresses the sum of f(x,y) along the line whose distance from the origin is s and whose normal vector is in θ direction. This sum, $g(s,\theta)$, is called ray-sum.

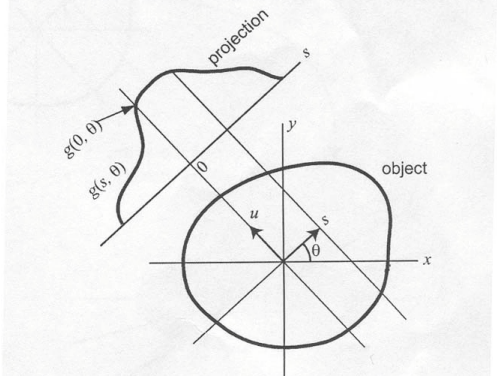


Fig. 1. Radon Transformation

The Radon transform could be computed for any angle, and could be displayed as an image. From Fig. 1, the Radon transform of the input images can be computed for any angle. In practice, we compute the Radon transform at angles from 0 to 179 degree, in 1 degree increment. The procedure to find the straight lines using the radon transform is as follows:

- Compute the binary edge image of input image using the edge detector
- Compute the Radon transform of the edge image at angles from 0 to 179
- Find the locations of strong peaks in the Radon transform matrix. Radon transform is a 2D matrix. The highest values in the matrix correspond to the strong peaks in the Radon transform. The location of these peaks corresponds to the location of straight lines in the original image.

The straight lines are drawn from the information obtained through the strong peaks in the Radon transform.

3 Detecting Corners Using Inverse Radon Transform

Ideally, a corner is an intersection of two straight lines. However, in practice, corners in the real world are frequently deformed with ambiguous shapes. As corner represent

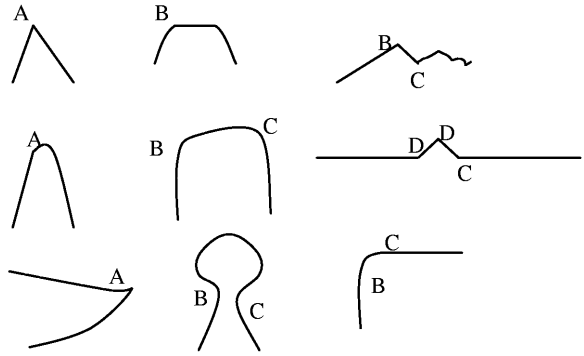


Fig. 2. Four types of corners

certain local graphic features at abstract level, corners can intuitively be described by some semantic patterns (see Fig. 2). A corner can be characterized as one of the following four types [17]:

- Type A: A perfect corner, i.e., a sharp turn of curve with smooth parts on both sides.
- Type B: The first of two connected corners similar to the END or STAIR models in, i.e., a mark of change from a smooth part to a curved part.
- Type C: The second of two connected corners, i.e., a mark of change from a curved part to a smooth part.

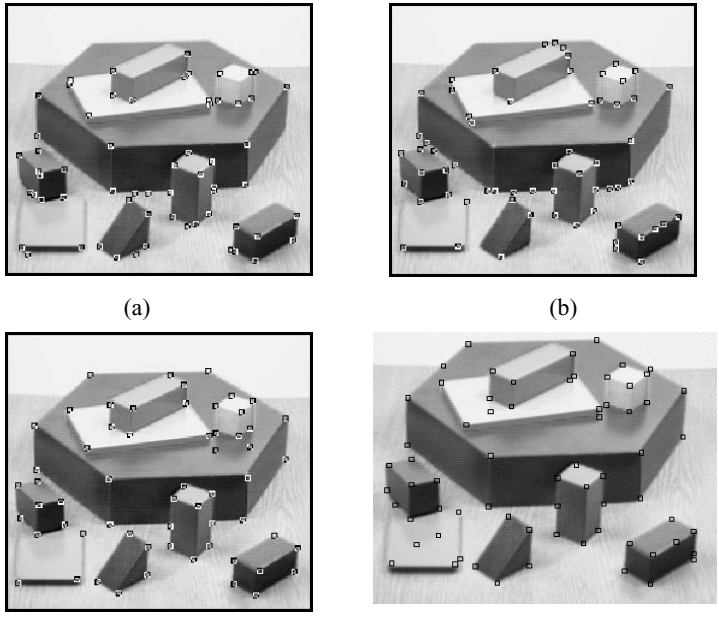


Fig. 3. Blocks image. (a) Kitchen/Rosenfeld. (b) SUSAN. (c) CSS. (d) Proposed Algorithm.

- Type D: A deformed model of type A, such as a round corner or a corner with arms neither long nor smooth. The final interpretation of the point may depend on the high level global interpretation of the shape.

Figure 2 shows some examples of the four types of the corner. It is obvious from the Fig.2 that the corner points at very small level are the intersection points of the two straight lines. To detect corner points in an image, we do the following procedure using the inverse Radon transform:

- Quantize (x, y) space into a two-dimensional array C of the original size of the image in appropriate steps of x and y.
- Initialize all elements of C(x, y) to zero.
- We draw the straight lines from the peaks of the Radon transform, and add 1 to all elements of C(x, y) whose indices x and y passed by the straight lines.
- Search for elements of C(x, y) which have large values than one. Each one found corresponds to a possible candidate for corners in the original image.

Because of many intersections of lines, false corners are also detected. To avoid false candidates, the detected corners whose vicinity does not contain any edge point are discarded. To remove the unwanted intersection points (i.e., no corner points), we ANDED the intersection points with the edges of the image. The position of true corners and the edges will coincide and give the actual position of the corners. To get more accurate results and to avoid large number of intersections, corners are detected

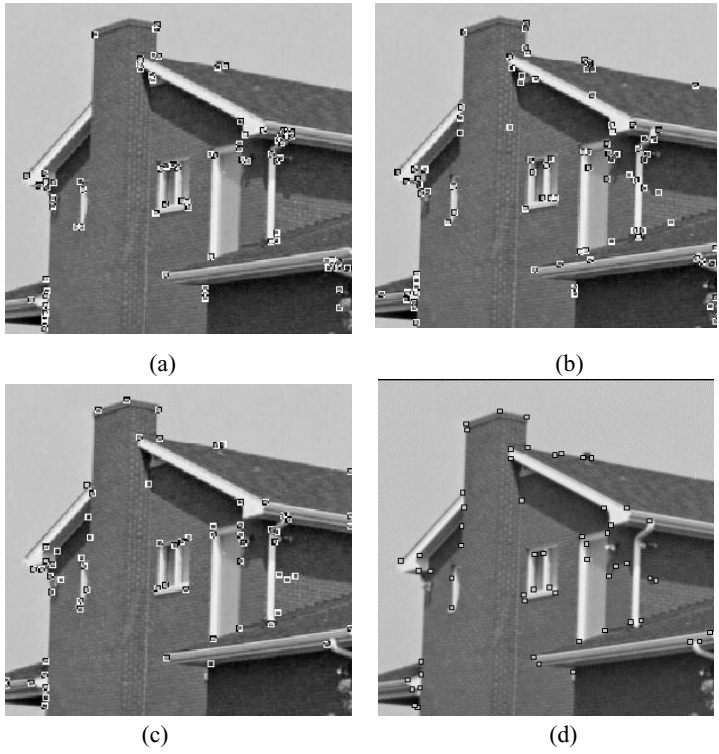


Fig. 4. House image. (a) Kitchen/Rosenfeld. (b) SUSAN. (c) CSS. (d) Proposed Algorithm.

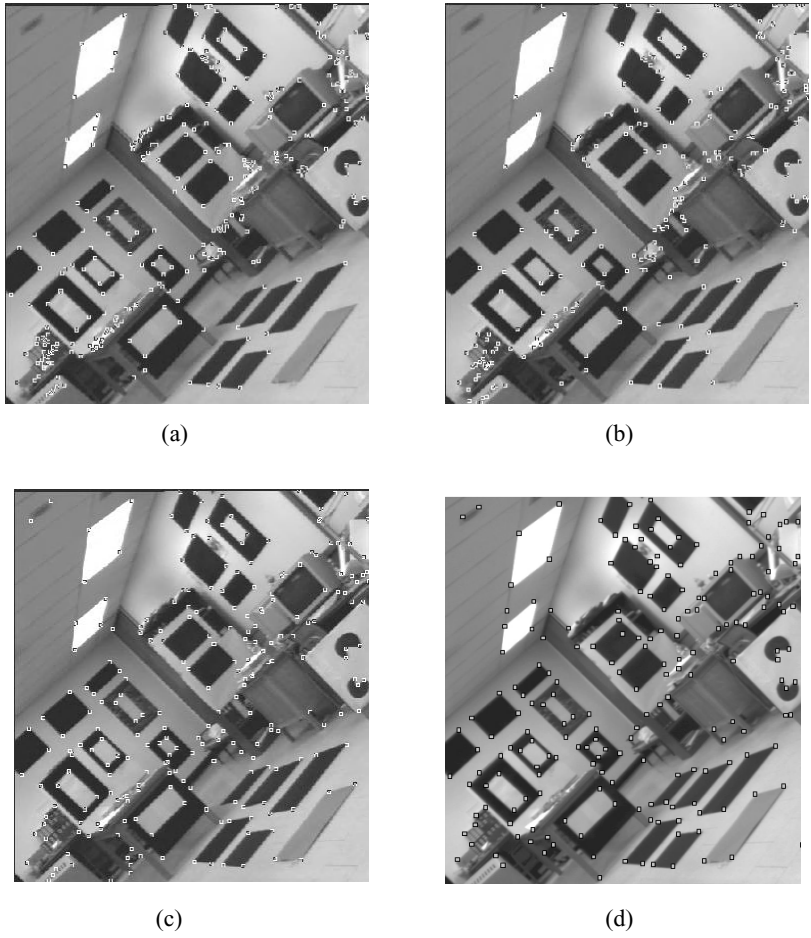


Fig. 5. Lab image. (a) Kitchen/Rosenfeld. (b) SUSAN. (c) CSS. (d) Proposed Algorithm.

block by block processing of input image with sliding overlapping window. The number of computations becomes higher, but the results will be improved.

4 Simulations

The proposed algorithm was tested using three different images, the same images used in [13]. The results are compared with three different corner detectors, namely, Kitchen and Rosenfeld (KR), SUSAN and CSS corner detectors. The results for the three corner detectors are also taken from [13]. The results for each detector were the best results obtained by searching the best parameters. The three test images are called Blocks, House and Lab. The Blocks test image contains much texture and

noise. The House image has a lot of small details and texture in the brick wall. The Lab image contains plenty of corners. The results show that the proposed algorithm gives better result than that of KR and SUSAN method and gives comparable results with that of CSS. The proposed method and CSS perform well on the Blocks image. Other detectors find difficulties in locating the corner points. Similarly, the proposed method and CSS method show good performance on the House and the Lab images, while others perform badly. Overall our proposed method and CSS are comparable.

5 Conclusions

In this paper, a new corner detector based on the Radon Transform is proposed. The edges in the image are found and the edges are transformed from the image space to parameter space. The Radon Transform is used to find the straight lines in the image, and the inverse Radon Transform is used to find the intersection points among the straight lines. The intersection points are the corner points. The proposed method is compared with the previous methods. The results are comparable with the curvature scale space image corner detector.

References

1. Liyuan Li and Weinan Chen, "Corner detection and interpolation on planar curves using fuzzy reasoning," IEEE Trans. On Pattern Analysis and Machine Vision, vol. 21, no. 11, November, 1999.
2. H. P. Moravec, "Towards automatic visual obstacle avoidance," Proc. Int'l Joint Conf. Artificial Intelligence, p. 584, 1977.
3. P. R. Beaudet, "Rotationally invariant image operators," Int'l Joint Conf. Pattern Recognition, pp. 579-583, 1978.
4. L. Kitchen and A. Rosenfeld, "Gray level corner detection," Pattern Recognition Letters, pp. 95-102, 1982.
5. K. K. Lai and P. S. Y. Wu, "Effective edge-corner detection method for defected images," Proc. Int'l Conf. Signal Processing, vol. 2, pp. 1,151-1,154, 1996.
6. Z. O. Wu and A. Rosenfeld, "Filtered projections as an aid to corner detection," Pattern Recognition, vol. 16, no. 31, 1983.
7. K. Paler, J. Foglein, J. Illingworth, and J. Kittler, "Local ordered grey levels as an aid to corner detection," Pattern Recognition, vol. 17, no. 5, pp. 535-543, 1984.
8. F. Arrebola, A. Bandera, P. Camacho, and F. Sandoval, "Corner detection by local histograms of contour chain code," Electronics Letters, vol. 33, no. 21, pp. 1,769-1,771, 1997.
9. K. Kohlmann, "Corner detection in natural images based on the 2-D Hilbert Transform," Signal Processing, vol. 48, no. 3, pp. 225-234, 1996.
10. S. M. Smith and J. M. Brady, "SUSAN—A new approach to low level image processing," Defense Research Agency, Technical Report no. TR95SMS1, Farnborough, England, 1994.
11. R. Mehrotra, S. Nichani, and N. Ranganathan, "Corner detection," Pattern Recognition, vol. 23, no. 11, pp. 1,223-1,233, 1990.
12. E. R. Davies, "Application of the generalized Hough Transform to corner detection," IEE Proc., vol. 135, pp. 49-54, 1988.

13. Farzin Mokhtarian and Riku Suomela, “Robust image corner detection through curvature scale space,” IEEE Trans. On Pattern Analysis and Machine Vision, vol. 20, no. 12, December, 1998.
14. F. Mokhtarian and A.K. Mackworth, “A theory of Multi-Scale, Curvature-based shape representation for planar curves,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 14, no. 8, pp. 789-805, Aug. 1992.
15. S. R. Deans, “Hough Transform from the Radon Transform,” IEEE Trans. On Pattern Analysis and Machine Vision, vol. 3, pp. 185-188, 1981.
16. S. R. Deans, The Radon Transform and some of its applications, Kreiger, 1983.
17. A. Rattarrangsi and R. T. Chin, “Scale-based detection of corners of planar curves,” IEEE Trans. On Pattern Analysis and Machine Vision, vol. 14, no. 4, pp. 430-449, 1992.

Analytical Comparison of Conventional and MCDF Operations in Image Processing

Yinghua Lu[1,2] and Wanwu Guo[3]

[1] Department of Computer Science, Northeast Normal University
138 Renmin Street, Changchun, Jilin, China
luyh@nenu.edu.cn
[2] School of Computer Science, Jilin University
Changchun, Jilin, China
[3] School of Computer and Information Science, Edith Cowan University
2 Bradford Street, Mount Lawley, Western Australia 6050, Australia
w.guo@ecu.edu.au

Abstract. Modified conjugate directional filtering (MCDF), a new method proposed by Guo and Watson in 2002 for digital data and image processing, provides ability in not only integrating directional-filtered results in conjugate directions into one image that shows the maximum linear features in these conjugate directions, but also further manipulating the outcomes using a number of predefined MCDF operations for different purposes. Although a number of cases have been used to test the usefulness of several proposed MCDF operations, and the results are 'visually' better than some conventional methods, however, no quantified analytical comparisons on its effectiveness over conventional methods have been obtained. In this paper, we firstly outline a FFT-based scheme for making analytical comparisons between the results from using conventional and MCDF operations, and then apply this scheme to an aerial photograph of a city view for making comparisons between directional-filtered and MCDF(add2) images. The comparison verifies that the MCDF(add2) operation indeed performs better than conventional directional filtering in image processing in terms of information integration and retention of low-frequency components.

1 Introduction

Guo and Watson [1] proposed a new method called the modified conjugate directional filtering (MCDF). Tests have shown that MCDF can integrate directionally filtered results in conjugate directions into one image that shows the maximum linear features in these conjugate directions, which cannot be achieved by using conventional directional filtering [2]. MCDF also enables further manipulation during the integration by using a number of predefined MCDF operations for different purposes [2]. MCDF is modified from the previous proposal named conjugate directional filtering (CDF) [3], because further study reveals that the CDF has two weaknesses, i.e., a weighting system for further data manipulation during the operation was not considered, and the CDF-processed images often lack contrast depth because most background information is removed as a result of applying directional filtering.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 956–963, 2004.
© Springer-Verlag Berlin Heidelberg 2004

MCDF overcomes these weaknesses by superimposing the weighted CDF image onto the original image. In this way, not only further enhanced by a weighting factor are the conjugated features, but also retained is all the information on the original image. By introducing these two measures into the CDF, MCDF becomes much more powerful [1][2].

Although a number of tests have shown the usefulness of several proposed MCDF operations, and the results are 'visually' better than some conventional methods [4][5], however, no quantified analytical comparisons on its effectiveness over conventional methods in linear enhancement have been done. Our new study on analytical tests on MCDF operations has resulted in the acquisition of some positive and encouraging outcomes. In this paper, we first briefly present the concepts of the MCDF operations, and then outline the scheme for making analytical comparisons between conventional methods and MCDF operations. The results of applying this analytical comparison to an image of city view using MCDF(add2) and conventional directional filtering are then presented in this paper.

2 Concepts of MCDF Operations

Assuming f_0 to be the original data file, f_1 and f_2 to be the directional-filtered data files in the two conjugate directions, the general expression of the MCDF operations can be written as [1]

$$\text{MCDF} = F_0(W_0 * f_0) + F_2[W_1 * F_1(f_1),\ W_2 * F_1(f_2)]; \tag{1}$$

where W_0, W_1 and W_2 are selective constants; F_0, F_1 and F_2 are pre-defined functions. By adjusting these parameters, various MCDF operations can be defined. For example, when we choose $F_0 = F_1 = F_2 = 1$ and addition operation, formula (1) becomes

$$\text{MCDF(add1)} = W_0 * f_0 + W_1 * f_1 + W_2 * f_2. \tag{2}$$

Its function is to simply combine the original image and two directionally filtered images in two conjugate directions together using different weights. When we choose $F_0 = 1$, $F_1 = 1$, $F_2 = \text{abs}$ and addition operation, formula (1) becomes

$$\text{MCDF(add2)} = W_0 * f_0 + \text{abs}(W_1 * f_1 + W_2 * f_2). \tag{3}$$

Its function is to combine the original data and the absolute value of the two directionally filtered data in two conjugate directions together using different weights. When choosing $F_0 = 1$, $F_1 = \text{abs}$, $F_2 = 1$ and using addition operation, formula (1) becomes

$$\text{MCDF(add3)} = W_0 * f_0 + W_1 * \text{abs}(f_1) + W_2 * \text{abs}(f_2). \tag{4}$$

3 Design of the Analytical Comparisons

Directional filtering is used to enhance linear features in a specific direction [4][5][6]. In some cases, identifying conjugate linear information on an image is particularly concerned. Directional filtering can be made in two specific conjugate directions to

enhance these conjugate features. Normally the filtered results from the two conjugate directions are shown on two separate images. This is inconvenient for revealing the relationships between linear features in these two conjugate directions. The linear enhancement using directional filtering is achieved by constraining or removing the textural features or low-frequency components from the original image to outline the structural features or high-frequency components contained in the original image. Thus, directionally filtered image often lacks contrast depth because most background information is removed.

These two weaknesses of using the conventional directional filtering are overcome by MDCF method, which firstly combines two (or more) directional-filtered results in conjugate directions into one image that exhibits the maximum linear features in these two conjugate directions, and secondly retains the background information by superimposing the directionally filtered data onto the original data. Therefore, the analytical comparison should be designed in a way through which these two improvements can be clearly revealed.

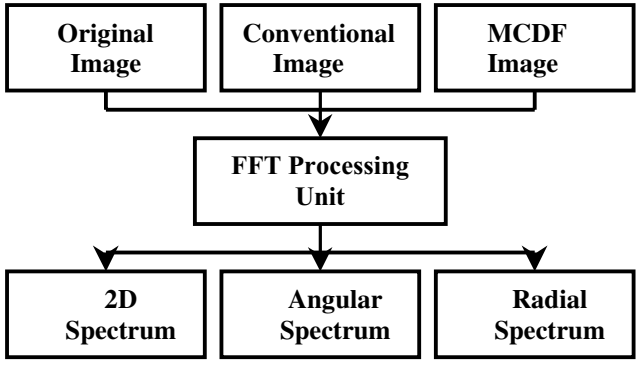


Fig. 1. Schematic diagrams of the design for analytical comparison of images

We propose a design for taking this analytical comparison shown in Figure 1. Firstly, original image and each of its processed images are sent individually to a processing unit for analysis using fast Fourier transform (FFT). The outcomes from this FFT analysis include a 2D Cartesian spectrum, an angular spectrum and a radial spectrum of the corresponding input image [7]. Compared with the outcomes of the original image, the 2D Cartesian spectrum is used to directly identify whether the MCDF operations indeed have brought enhanced information in the conjugate directions into MCDF-processed images; the angular spectrum should be able to reveal the effectiveness of different operations applied to the original image; the radial spectrum is used to quantify whether the MCDF-processed images have retained the background information or low-frequency components of the original image while the structural features or high-frequency components are enhanced.

To make the analytical results acceptable as widely as possible, the FFT analysis is carried out using FFT functions provided by Matlab [8][9]. Next section reports the test results of MCDF(add2) operation on a photograph of a city view.

4 Analytical Results of Aerial Photograph of City View

Figure 2a is the original aerial photograph over a small part of a city. There are buildings, streets, sport courts, trees, cars on parking areas and other objects on the image. Most of these objects are arranged in either the north-south direction or the east-west direction, or in the both directions. To enhance the north-south trending features in this photograph, a conventional Sobel horizontal operator is applied to this image and the result is shown in Figure 2b. Result of using Sobel vertical operator is illustrated in Figure 2c. In both images, directional filtering in the NS and EW directions has enhanced the linear features along their conjugated directions, but images look very dark because the low-frequency components in the original photograph has been removed or depressed.

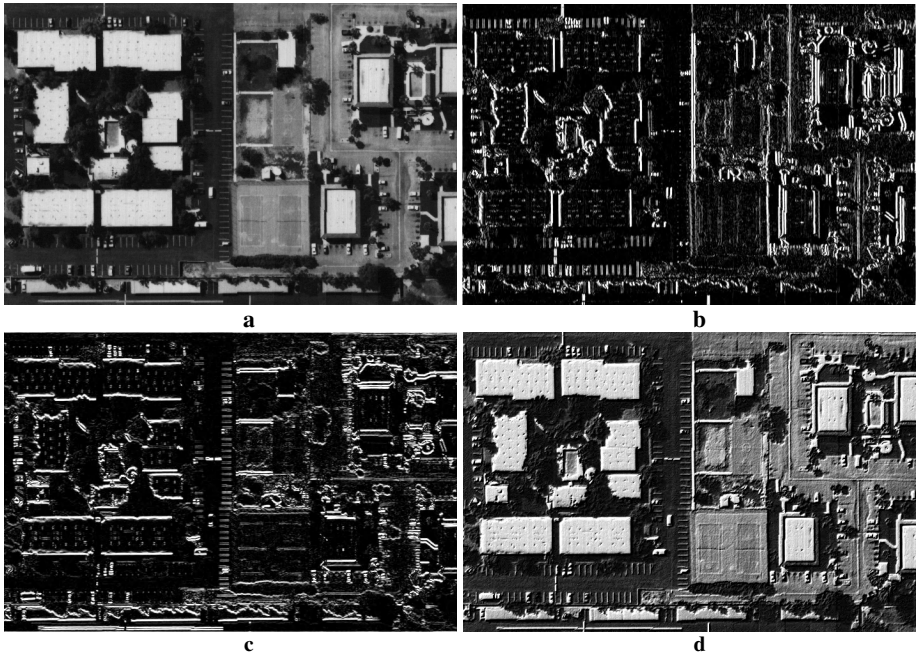


Fig. 2. Original (a), Sobel horizontal (b), Sobel vertical (c), and MCDF(add2) (d) images

Figure 2d shows the image after applying MCDF(add2) with $W_0 = 1$ and $W_{NS} = W_{EW} = 2$ to the original photograph. This MCDF(add2) operation enhances features in both the EW and NS directions, e.g., the blurry conjugate marks in the tennis courts are clearly outlined. Unlike Sobel images, MCDF(add2) image also keeps the background information.

Figure 3a shows the 2D Cartesian spectrum of the original image. The conjugated NS-EW trending features are reflected as alignments along both the horizontal and vertical axes. It should be noticed that axes in frequency domain are opposite to those in spatial domain, i.e., vertical and horizontal axes in 2D spectrum image correspond to horizontal and vertical directions in the original photograph. Figure 3b shows that Sobel horizontal operator has indeed enhanced vertical components, but this is

achieved by depressing the horizontal components. In contrast, Sobel vertical operator removes the vertical components to boost the horizontal information (Fig. 3c).

Figure 3d shows the spectrum of the MCDF(add2) image. Like the spectrum of original image, the conjugated NS-EW trending features are clearly reflected as clusters along both axes. However, the area of the frequency clusters in Figure 3d is clearly larger than that in Figure 3a, which indicates that this MCDF operation has not only enhanced the features in the conjugated NS-EW directions, but increased the overall percentage of the high-frequency components in the image.

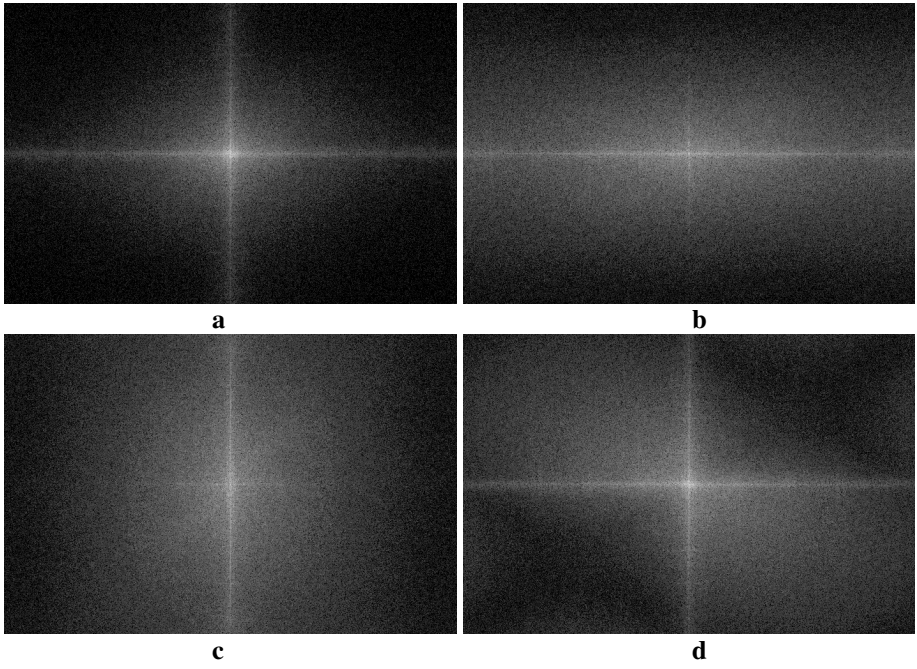


Fig. 3. 2D spectrum of original (a), Sobel horizontal (b), Sobel vertical (c), and MCDF(add2) (d) images

Figure 4a shows the 2D angular spectrum of the original image. Frequency components are clustered along the conjugated NS-EW directions. Figure 4b shows that Sobel horizontal operator has indeed enhanced vertical components, but the horizontal components are reduced significantly. Sobel vertical operator almost removes all the vertical components in order to enhance the horizontal information (Fig. 4c).

Figure 4d shows the angular spectrum of the MCDF(add2) image. The conjugated NS-EW trending features are clearly reflected as clusters along both directions, but the high-frequency components are much elevated than its original level (Fig. 4a), in particular along the NE-SW direction. This is because the two conjugate directional filters are moved from south to north and from west to east, respectively. MCDF(add2) combines these two data into one image, which mathematically composites a SW-NE trending vector, and consequently enhances alignments in SW-NE direction. This feature is also reflected in Figure 3d where a NW-SE trending zone in frequency domain is formed.

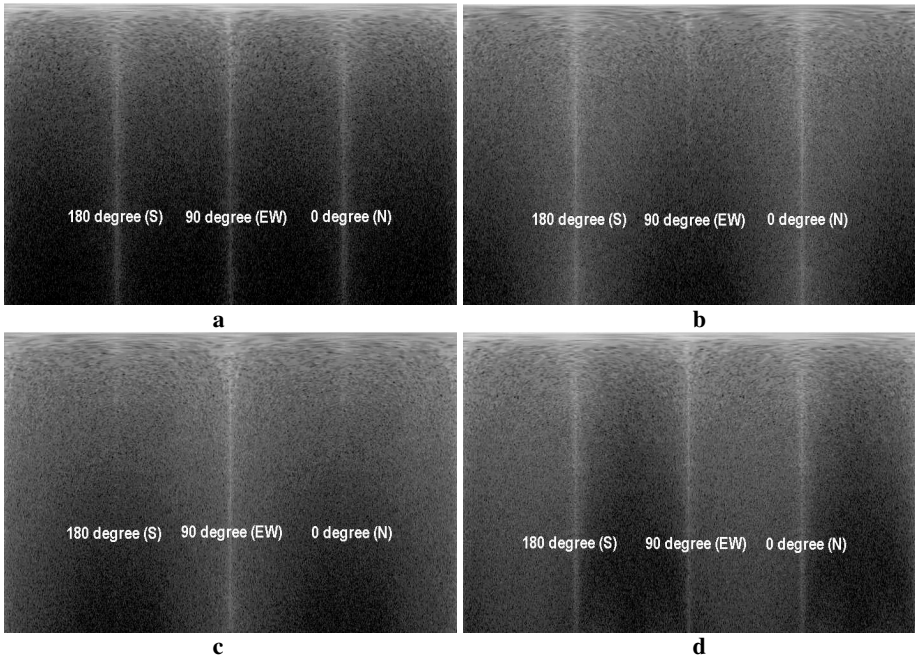


Fig. 4. Angular spectrum of original (a), Sobel horizontal (b), Sobel vertical (c), and MCDF(add2) (d) images

Figure 5a shows the intensity of different frequency components in the original image. At frequency of 20 Rad/s, the intensity is ~75% of the maximum intensity. In radial spectrum diagrams of both Sobel horizontal and vertical images (Figs. 5b & 5c), this ratio is reduced to ~60%. However, in the radial spectrum of MCDF(add2) image, a ratio of 75% is maintained. This implies that the MCDF(add2) operation indeed retains the background information of the original image, unlike the conventional directional filtering that depresses the low-frequency components in exchange for enhancing higher frequency components.

This is further illustrated in the statistics of the original and MCDF(add2) spectra (Table 1). The MCDF(add2) operation has enhanced the highest-frequency component by 2.5 times from its relative intensity of 7.8% in the original image to 20.9% in the MCDF(add2) image. This is achieved by keeping almost no change in the maximum intensity in both images, which means that there is almost no loss in low-frequency components in the MCDF(add2) image. The medium-frequency components are also intensified from 23.2% in the original image to 33.5% in the MCDF(add2) image. By keeping the same low-frequency components, bringing a moderate increase in medium-frequency components, and elevating high-frequency components by at least 2.5 times, all together the MCDF(add2) operation makes not only features in the NS and EW directions in the MCDF(add2) image look more prominent, but also the whole image appear richer in contrast depth and thus more smooth.

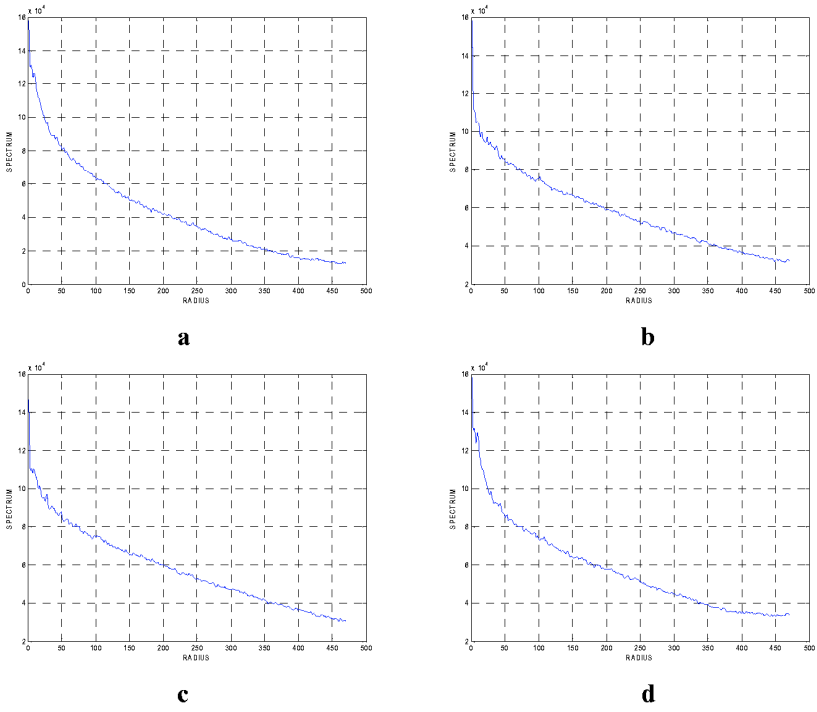


Fig. 5. Radial spectrum of original (a), Sobel horizontal (b), Sobel vertical (c), and MCDF(add2) (d) images

Table 1. Statistics of radial spectra of Figure 2a and its MCDF(add2) image

Statistics	Original image		MCDF(add2) image	
	Absolute	Relative (x/Max)	Absolute intensity	Relative (x/Range)
Min (high-frequency components)	12297	7.8%	32974	20.9%
Max (low-frequency components)	158111	100%	158105	100%
Median (medium-frequency components)	36711	23.2%	53042	33.5%

Table 2. Statistics of radial spectra of the original digital terain model (DTM) and its MCDF(add1) image

Statistics	Original image		MCDF(add1) image	
	Absolute	Relative (x/Max)	Absolute intensity	Relative (x/Range)
Min (high-frequency components)	826	0.5%	7446	4.5%
Max (low-frequency components)	164359	100%	164345	100%
Median (medium-frequency components)	10372	6.3%	27810	16.9%

Although the results of using MCDF(add2) are presented here only, tests over other MCDF operations also reveal the similar results (Table 2) [10].

5 Conclusion

Our FFT analytical comparison between conventional directional filtering and MCDF operation on the aerial photograph proves that the MCDF(add2) operation brings enhanced information integration and retains background information whereas the higher-frequency components are enhanced, which are the weaknesses of using the conventional methods in image processing. Although the results of using MCDF(add2) are presented here only, tests on other MCDF operations also reveal the similar results. Therefore, the MCDF method is effective and worth for further development.

Acknowledgements. We are grateful to Faculty of Computing, Health and Science of the Edith Cowan University for supporting this research project. The Northeast Normal University is thanked for supporting Dr W Guo's visit to its computer science department. We appreciate the assistance provided by Dr J Kong in data preparation. The constructive comments made by the anonymous referees are acknowledged.

References

1. Guo, W., Watson, A.: Modification of Conjugate Directional Filtering: from CDF to MCDF. Proceedings of IASTED Conference on Signal Processing, Pattern Recognition, and Applications. Crete, Greece (2002) 331-334.
2. Guo, W., Watson, A.: Conjugated Linear Feature Enhancement by Conjugate Directional Filtering. Proceedings of IASTED Conference on Visualization, Imaging and Image Processing. Marbella, Spain (2001) 583-586.
3. Watson. A., Guo, W.: Application of Modified Conjugated Directional Filtering in Image Processing. Proceedings of IASTED Conference on Signal Processing, Pattern Recognition, and Applications. Crete, Greece (2002) 335-338.
4. Jahne, B.: Digital Image Processing: Concepts, Algorithms and Scientific Applications. Springer-Verlag, Berlin Heidelberg (1997).
5. Proakis, J.G., Manolakis, D.G.: Digital Signal Processing: Principles, Algorithms and Applications. Prentice-Hall, Upper Saddle River New York (1996).
6. Richards, J.A.: Remote Sensing Digital Image Analysis. Springer-Verlag, Berlin Heidelberg (1993).
7. Gonzalez, R.C., Woods, R.E.: Digital Image Processing, Prentice Hall (2002).
8. Hanselman, D., Littlefield, B.R.: Mastering MATLAB 6. Prentice Hall (2001).
9. Phillips, C.L., Parr, J.M., Riskin, E.A.: Signals, Systems, and Transforms. Prentice Hall (2003).
10. Kong, J., Zhang, B., Guo, W.: Analytical test on effectiveness of MCDF operations. Computational Science, Lecture Notes in Computer Science, Springer-Verlag, Berlin Heidelberg (2004).

On Extraction of Facial Features from Color Images

Jin Ok Kim[1], Jin Soo Kim[2], Young Ro Seo[2], Bum Ro Lee[2],
Chin Hyun Chung[2], Key Seo Lee[2], Wha Young Yim[2], and Sang Hyo Lee[2]

[1] Faculty of Multimedia, Daegu Haany University,
290, Yugok-dong,Gyeongsan-si, Gyeongsangbuk-do, 712-715, KOREA
bit@dhu.ac.kr
[2] Department of Information and Control Engineering, Kwangwoon University,
447-1, Wolgye-dong, Nowon-gu, Seoul, 139-701, KOREA
chung@kw.ac.kr

Abstract. Human face detection is one of the most important processes in applications such as video surveillance, human computer interface, face recognition, and image database management. Algorithms have been discussed in lots of papers about face detection and face recognition. But it is well known that their implementation is not easy. Due to variations in illumination, background, visual angle and facial expressions, the problem of machine face detection is complex. Face detection algorithms have primary factors that decrease a detection ratio: variation by lighting effect, location and rotation, distance of object and complex background. We propose a face detection algorithm for color images in the presence of varying lighting conditions as well as complex background. We use the YC_bC_r color space since it is widely used in video compression standards and multimedia streaming services. Our method detects skin regions over the entire image, and then generates face candidate based on the spatial arrangement of the skin patches. The algorithm constructs eyes, mouth, nose, and boundary maps for verifying each face candidate.

1 Introduction

Human activity is a major concern in a wide variety of applications such as video surveillance, human computer interface, face recognition, and face image database management. And machine face recognition is a research filed of fast increasing interest. Although a lot of work has already been done, a robust extraction of facial regions and features out of complex scenes is still a problem [1]. In the first step of face recognition, the localization of facial regions and the detection of facial features, e.g., eyes and mouth, are necessary. Detecting face is a crucial step in the identification applications. Most face recognition algorithms assume that the face location is known. Similarly, face tracking algorithms often assume that the initial face location is known. Note that face detection can be viewed as a two-class (face versus non-face) classification problem. Therefore, some techniques developed for face recognition (e.g., holistic/template approaches, feature based approaches, and their combination) have also been used

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 964–973, 2004.
© Springer-Verlag Berlin Heidelberg 2004

to detect faces, but they demand much computational process and cannot handle large variations in face images. Various approaches to the face detection are discussed in [2,3,4,5,6]. For recent surveys on the face detection, one can refer to [3,6]. These approaches utilize techniques such as principal component analysis, neural networks, machine learning, information theory, geometrical modeling, (deformable) template matching, Hough transform, motion extraction, and color analysis. The neural network-based and view-based approaches require a large number of face and non-face training examples and are designed primarily to locate frontal faces in gray-scale images. Schneiderman and Kanade extend their learning-based approach for the detection of frontal faces to profile views. A feature-based approach that uses geometrical facial features with belief networks provides the face detection for non-frontal views. Categorizing face detection methods based on the representation used reveals that detection algorithms using holistic representations have the advantage of finding small faces or faces in poor-quality images, while those using geometrical facial features provide a good solution for detecting faces in different poses. A combination of holistic and feature-based approaches is a promising approach to face detection as well as face recognition. Skin-tone color are useful cues for face detection. However, it is difficult for color-based approaches to detect robustly skin colors in the presence of complex background and different lighting conditions. We propose a face detection algorithm that is able to handle a compensation technique. We construct a skin color model in YC_bC_r color space and suggest the lighting compensation algorithm for variative light condition. Also we get the facial candidate region using pixel connectivity by morphological operation. At last we determine the feature point in color image.

2 Face Region Detection Algorithm

An overview of our face detection algorithm is depicted in Fig. 1, which contains two major modules. They are face segmentation for finding face candidates and facial feature extraction for verifying detected face candidates. Our approach for face localization is based on the observation that human faces are characterized by their oval shape and skin color, also in the case of varying light conditions. Therefore, we locate face-like regions on the base of shape and color information. We employ the YC_bC_r color space by using the RGB to YC_bC_r transformation. The hypotheses for faces are verified by searching for facial features inside the facial regions. We extract facial features based on the observation that eyes and mouth differ from the rest of the face in chrominance because of their conflictive response to C_b, C_r. We consider chrominance and luminance as discriminating color information. These attributes are strongly related to the human perception of color. To obtain robustness against changes in lighting conditions, we perform region segmentation by considering appropriately defined domains in chrominance and luminance domain. These domains can be estimated a priori and used subsequently as reference for any skin color.

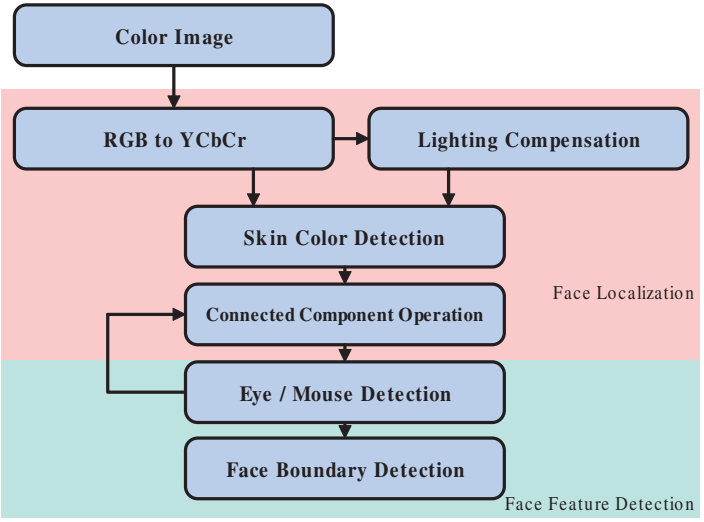


Fig. 1. Face detection algorithm

This algorithm reduces error ratio using lighting compensation process that over exposure. With compensated RGB image transformed by YC_bC_r color model, we present skin color model using luma-independent C_bC_r model.

2.1 Light Compensation Algorithm

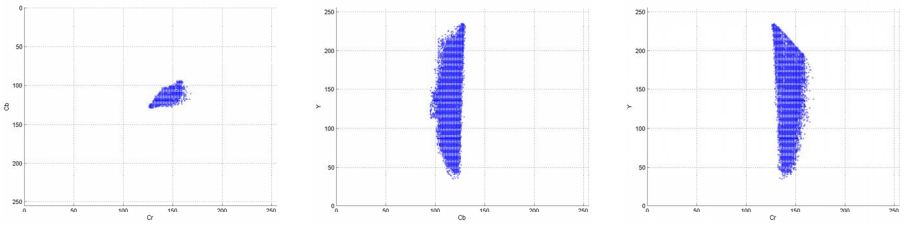
Skin-tone color depends on the lighting conditions. We introduce a lighting compensation technique that use "reference white" to normalize the color appearance. We regard pixels within top 5 percent of the luminance values in the image as the reference white only if the number of these pixels is sufficiently large [1].

- Calculate luminance range in entire image.
- Define reference white pixels belonging to top 5 percent of the luminance values in the image.
- If reference white pixels are bigger than threshold level, then scale R,G,B values

2.2 Skin Tone Detection

Color is a powerful fundamental cue that can be used as a first step in the process of face detection in complex scene images because color image segmentation is computationally fast while being relatively robust to changes in illumination, in viewpoint, in scale, to shading and to complex (cluttered) backgrounds, compared with the segmentation of grey-level images. Modeling skin color requires choosing an appropriate color space and identifying a cluster associated with

skin color in the space. It has been observed that the normalized red-green (rg) space is not the best choice for the face detection. Based on Terrillon et al.'s [7] comparison of nine different color spaces for face detection, the tint-saturation-luma (TSL) space provides the best results for two kinds of Gaussian density models (unimodal and a mixture of Gaussians). We adopt the YC_bC_r space since it is perceptually uniform, it is widely used in video compression standards (e.g., MPEG and JPEG), and it is similar to the TSL space in terms of the separation of luminance and chrominance as well as the compactness of the skin cluster. Many researchs assume that the chrominance components of the skin-tone color are independent of the luminance component [8,9,10]. However, in practice, the skin-tone color is nonlinearly dependent on luminance. We demonstrate the luma dependency of skin-tone color in different color spaces in Fig. 2, based on skin patches collected from IMDB in the Intelligent Multimedia Laboratory [11].



(a) (b) (c)

Fig. 2. Distribution of skin tone pixel: (a) Cb vs Cr (b) Y vs Cb (c) Y vs Cr

We consider the area restricted by face mask enclosing the grouped skin-tone regions with a connected component labeling. Fig. 3 shows an example of the face mask.

2.3 Determine the Eye Map and Mouth Map

We first build two separate eye maps; one from the chrominance components and the other from the luminance components. Two maps are then combined into a single eye map. The eye map from the chroma is based on the observation that high C_b and low C_r values are found around the eyes [1]. It is constructed by

$$C_{eye} = \frac{1}{3}\{(C_b^2) + (\tilde{C}_r)^2 + (C_b/C_r)\} \tag{1}$$

where C_b^2, $(\tilde{C}_r)^2$, and $\frac{C_b}{C_r}$ are normalized to the range [0,255] and $\tilde{C}_r$ is the negative of C_r. Since the eyes usually contain both dark and bright pixels in

(a) (b)

Fig. 3. (a) Original image (b) Skin tone extraction

the luminance component, gray-scale morphological operator (e.g, dilation and erosion) can be designed to emphasize brighter and darker pixels in the luminance component around eye region. We use gray-scale dilation and erosion with a hemispheric structuring element to construct the eye map from the luminance as follows:

$$L_{eye} = \frac{Y(x,y) \oplus g(x,y)}{Y(x,y) \ominus g(x,y) + 1} \tag{2}$$

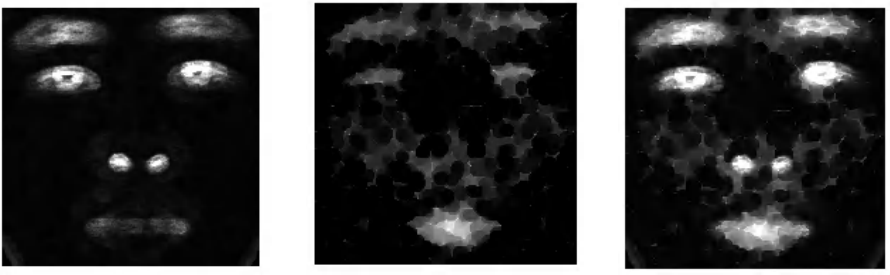
where the gray-scale dilation $\oplus$ and erosion $\ominus$ operations on a function $f : F \subset R^2 \to R$ using a structuring function $g : G \subset R^2 \to R$ are defined in [12]. The eye map from the chroma is enhanced by histogram equalization and then combined with the eye map from the luminance by an AND (multiplication) operation, i.e., $EyeMap = (EyeMapC)$ AND $(EyeMapL)$.

The color of mouth region contains stronger red component and weaker blue component than other facial regions[1]. Hence, the chrominance component C_r is greater than C_b in the mouth region. We further notice that the mouth has a relatively low response in the $\frac{C_r}{C_b}$ feature, but it has a high response in ${C_r}^2$. We construct the mouth map as follows:

$$Map_{mouth} = {C_r}^2 \times ({C_r}^2 - \eta \times \frac{C_r}{C_b})^2 \tag{3}$$

$$\eta = 0.95 \times \frac{\frac{1}{n} \sum_{(x,y) \in FG} C_r(x,y)^2}{\frac{1}{n} \sum_{(x,y) \in FG} \frac{C_r(x,y)}{C_b(x,y)}} \tag{4}$$

where both C_r^2 and $\frac{C_r}{C_b}$ are normalized to the range [0,255], and n is the number of pixels within the face mask, FG. The parameter η is estimated as a ratio of the average C_r^2 to the average $\frac{C_r}{C_b}$.



(a) (b) (c)

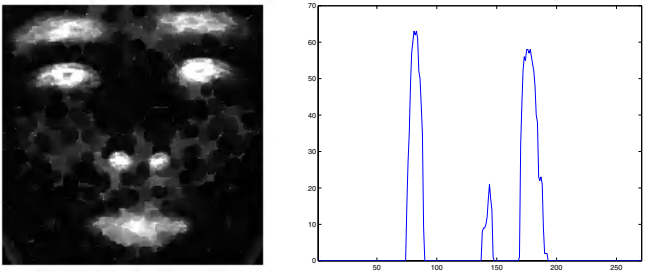
Fig. 4. (a) Eye map (b) Mouth map (c) Final map image

Fig. 4(b) shows the construction of the mouth map for the subject in Fig. 4(a)

2.4 Extraction Facial Features

Published feature detection methods can be generally summarized as following. (1) Image analysis based on geometry and gray-level distribution. This is the most intuitional approach. It aims to find the points or structures on the images with distinct geometrical shapes or gray-level distributions. (2) Deformable template matching. Originally worked out by Yille et al., this method has been utilized and improved by many researchers. (3) Algebraic feature extraction. Wavelet analysis and artificial neural network are also used for facial feature detection. Region description by projections is usually connected to binary image processing. Projections can serve as a basis of definition of related region descriptors; for example, the width (height) of a region with no holes is defined as the maximum value of the horizontal (vertical) projection of a binary image of the region. Afterwards, facial features are extracted by the analysis of the minima and the maxima. This is the reason why we evaluate the projections of the topographic gray level relief of the connected components. First the y-projection is determined by computing the mean gray level value of every rows of the connected component. Then minima and the maxima are searched in the smoothed y-relief. By checking the gradient, significant maxima are selected. For each significant maximum of the y-relief, x-reliefs are computed by averaging the gray level values of 3 neighbored rows of every columns. After smoothing

the x-reliefs, the minima and the maxima are determined. Beginning with the uppermost maxima of the y-relief, we search through the list of minima and maxima of x-reliefs to find facial feature candidates. For example, in case of the eyes, we search for two maxima that meet the requirement for eyes concerning relative position inside of head, significance of maximum between minima, ratio of distance between maxima to head width and similarity on gray level values. In case of the mouth, we look for two maxima that form the border of the mouth region. As result, we obtain a set of facial feature candidates.



(a) (b)

Fig. 5. (a) Map image (b) Horizontal profile of final map image

Horizontal and vertical region projections $p_h(i)$ and $p_v(j)$ are defined by

$$p_h(i) = \sum_j f(i,j) \tag{5}$$

$$p_v(j) = \sum_j f(i,j) \tag{6}$$

The feature vector [13] is determined by

$$\|u\|\|v\| cos\theta = u \cdot v$$

$$cos\theta = \frac{u \cdot v}{\|u\|\|v\|}$$

The length (or norm) of v(feature vector) is the nonnegative scalar $\|v\|$ defined by

$$\|v\| = \sqrt{v \cdot v} = \sqrt{v_1^2 + v_2^2 + \cdots + v_n^2},$$

and

$$\|v\|^2 = v \cdot v$$

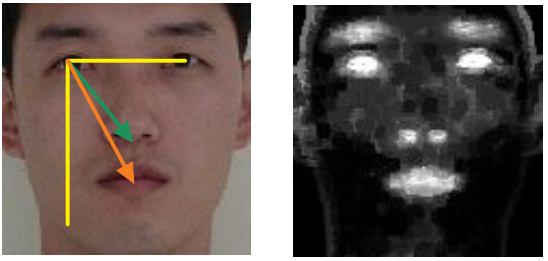


Fig. 6. Extraction of feature vector

In our search for features, we might try to capitalize on the observation that mouth is farther from right eye than nose. Now we have two features of the angle x_1 and the length x_2 [14].

3 Experimental Result

More than 100 images have been tested by using the presented algorithm. The test platform is a P4/2.4GHz computer with 512MB RAM under Windows 2K. We implement the face region detection process using YC_bC_r color space. As method to find eye and mouth, we use the response to a chrominance component. There are some restrictions in a general image like digital photos (e.g., complex background, and variation of lighting condition). Thus it is a difficult process to determine skin-tone's special features and to find the location of the eye and the mouth. Nevertheless we can produce an efficient algorithm that is robust in variation of lighting condition to use chrominance component in YC_bC_r color space. Also we can remove the fragment regions by using morphological process and connected component labeling operation. We find the location of eye and mouth by using vertical and horizontal projections. This method is useful and shows that operation speed is fast.

4 Conclusion

We have presented an approach that detects facial regions in color images and calculates facial features. Facial parts are determined on the basis of color and shape information. Therefore, we first extract regions with skin-like chrominance and luminance values and then compute the best fit ellipse for each of the regions. On the base of the observation that eyes and mouth differ from the rest of the face because of their lower brightness and difference between responses to chrominance, we first enhance the eye and mouth regions inside of the ellipse by applying morphological operations. Then we determine the position of facial features by evaluating the horizontal and vertical projections and topographic

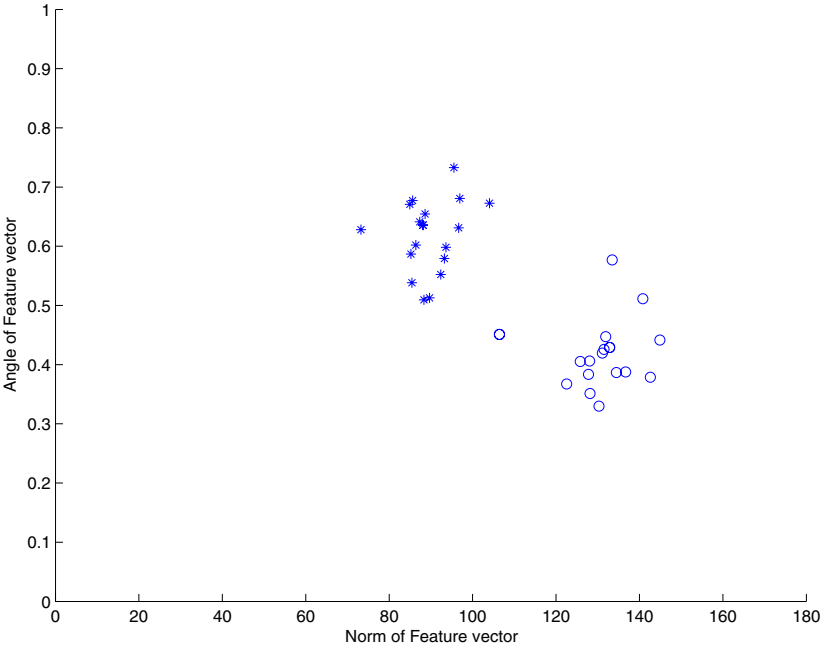


Fig. 7. Feature Space

grey level relief. The success of this method is verified on a large number of color images containing faces.

The difficulty on detecting facial features can be badly corrupted due to illumination, noise or occlusion. Skin tone extraction can be weakened by illumination. A truly robust system makes use of a luma-independent color space and features that should be invariant to different kinds of imaging conditions.

Our future work will be aimed at implementation of a real-time system for detecting facial features and tracking faces.

Acknowledgement. The present research has been conducted by the Research Grant of Kwangwoon University in 2004.

References

1. Hsu, R.L., Abdel-Mottaleb, M.: Face detection in color images. IEEE Pattern Analysis and Machine Intelligence **24** (2002) 696–706
2. Feraud, R., Bernier, O., Viallet, J.E., Collobert, M.: A fast and accurate face detection based on neural network. IEEE Trans. Pattern Analysis and Machine Intelligence **23** (2001) 42–53
3. Hjelmas, E., Low, B.: Face detection : A survey. Computer Vision and Image Understanding **83** (2001) 236–274

4. Maio, D., Maltoni, D.: Real-time face location on gray-scale static images. Pattern Recognition **33** (1999) 1525–1539
5. Pantic, M., Rothkrantz, L.: Automatic analysis of facial expressions : The state of the art. IEEE Trans. Pattern Analysis and Machine Intelligence **22** (1996) 1424–1445
6. Yang, M.H., Kreigman, D.J., Ahuja, N.: Detecting faces in images : A survey. Pattern Analysis and Machine Intelligence **24** (2002) 34–58
7. Terrillon, J.C., Akamatsu, S.: Comparative performance of different chrominance space for color segmentation and detection of human faces in complex scene images. Proc. IEEE Int'l Conf. on Face and Gesture Recognition (2000) 54–61
8. Menser, B., Brunig, M.: Locating human faces in color images with complex background. Intelligent Signal Processing and Comm. Systems (1999) 533–536
9. Saber, E., Tekalp, A.: Frontal-view face detection and facial feature extraction using color, shape and symmetry based cost functions. Pattern Recognition Letters **19** (1998) 669–680
10. Sobottka, K., Pitas, I.: A novel method for automatic face segmentation, facial feature extraction and tracking. Signal Processing : Image Comm. (1998) 263–281
11. IMDB: Designed, photographed, normalized and postprocessed by the members of intelligent multimedia lab. postech, korea. , http://nova.postech.ac.kr (2001)
12. Jackway, P., Deriche, M.: Scale-space properties of the multiscale morphological dilation-erosion. IEEE Trans. Pattern Analysis and Machine Intelligence **18** (1996) 38–51
13. Lay, D.C.: Linear Algebra And Its Applications 2nd Ed. Addison-wesley (1999)
14. Duda, R.., Hart, P.E., Stork, D.G.: Pattern Classification 2nd Ed. John Wiley & sons, New York (2001)

An Architecture for Mobility Management in Mobile Computing Networks

Dohyeon Kim[1] and Beongku An[2]

[1]School of Information and Communications, Cheonan University
Cheonan-City, Chungnam, Korea, 330-180, Tel.:041-620-9418
dhkim@cheonan.ac.kr
[2]School of Electronic, Electrical and Computer Engineering, Hongik University
Jochiwon, Chungnam, Korea, 339-701, Tel.:041-860-2243
beongku@wow.hongik.ac.kr

Abstract. In this paper, we present an efficient architecture for mobility management in mobile computing networks. The proposed architecture reduces the registration update cost and organize the physical location information databases in the lowest level into clusters, called logical groups, which are then represented by logical nodes in the next layer up. These logical nodes are regrouped into clusters and treated as single nodes in the next level up for all successive levels until all mobility agents are included in a tree-type logical hierarchy architecture. The performance evaluation of the proposed architecture is achieved via modeling analysis. The corresponding results demonstrate the proposed architecture can efficiently reduce the number of databases and average total registration cost.

1 Introduction

The proliferation of wireless LAN technologies and mobile nodes has prompted an increased need to support efficient and seamless roaming. Current mobility management protocols, such as the Mobile IP, as defined in RFC 2002, do not scale well into these requirements. Mobile IP employs mobility agents called home agent(HA) and foreign agent(FA) to support Internet-wide mobility [1-7]. However, when the distance between the foreign agent and the home agent is large, the signaling delay for the registration may be long, and results in long service disruption and packet losses. It is a major drawback of Mobile IP.

Some of the recent suggestions, notably celluar IP and HAWAII, use hierarchy of FA's for solving this drawback in the basic Mobile IP. Perkins and Wang proposed a solution that uses buffering with duplicate packet elimination to minimize the number of lost packets in the registrations. In addition, they used hierarchical FA management to reduce the overhead of frequent handoffs. They simulated the proposed solution and achieved substantial performance improvements. Forsberg and Malinen present a distribution of the mobility agent functionalities with fully scalable, arbitrarily deep tree hierarchies of foreign agents. In these suggestions, mobility agents use the physical hierarchy structure of foreign agents, this incurs database wastage and increases the cost of registration update [8-9].

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 974–982, 2004.
© Springer-Verlag Berlin Heidelberg 2004

In this paper, we present a new organizational method called the logical hierarchy architecture of foreign agents to reduce the registration update cost in mobile computing environments. The construction of the proposed logical hierarchy architecture of mobility agents starts with organizing the lowest level physical foreign agents into clusters called logical groups, which are then represented by logical nodes in the next layer up. Thereafter, these logical nodes are regrouped into clusters and treated as single nodes in the next level up for all successive levels until all foreign agents are included in a tree-type logical hierarchy architecture. As a result, the proposed logical hierarchy architecture can reduce the number of databases and thereby the average registration update cost.

The rest of the paper is organized as follows. Section 2 presents the proposed logical hierarchy architecture of foreign agents for mobile computing environments. Section 3 illustrates the registration update procedure using the proposed logical hierarchy architecture. Section 4 compares the performance of the proposed logical hierarchy architecture with the existing architecture in terms of the number of databases and total registration cost. Section 5 gives some final conclusions.

2 Logical Architecture for Mobility Management

In the existing mobility agent architecture, the foreign agents hold information on the identification of all terminals currently registered within the registration area. This information is used to setup a call. In case of the existing hierarchy architecture of foreign agents, the foreign agents in the upper level store the registered mobile node information in their foreign agent on a lower level. In contrast, the proposed hierarchy architecture of foreign agents uses a logical layer concept whereby the physical foreign agents only exist on the lowest level.

For efficient registration, the proposed logical hierarchy architecture of foreign agents forms hierarchical logical groups, as shown in Fig. 1. Starting on the lowest level, the physical foreign agents are bundled up into logical groups, which are then represented in the next layer up by logical nodes. These logical nodes are again grouped into clusters that are then treated as single nodes in the next layer up. So, the lower logical nodes are successively bundled up into upper logical groups until all the foreign agents are included in a tree-type logical hierarchy architecture.

In the proposed logical hierarchy architecture of foreign agents, the lowest foreign agents include the registration area, referred to as the lowest peer group, which manages the location information on the mobile nodes. Therefore, a logical group is formed to bundle these foreign agents, then a mobility agent group leader which plays the role of the mobility agent in the next level up. Accordingly, while the foreign agents in the lowest level have a physical existence, the foreign agents in the upper level logical groups are merely virtual.

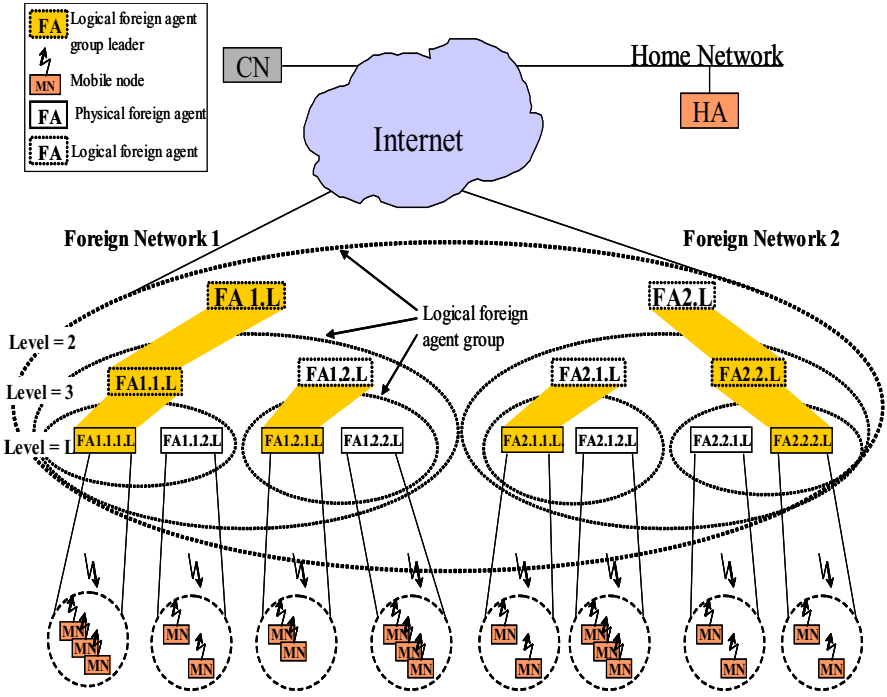


Fig. 1. Proposed logical hierarchy architecture of foreign agents

3 Registration Procedure in Logical Architecture

In the proposed logical hierarchy architecture of mobility agents, registration occurs when a mobile node moves across the boundary of the registration area, as seen in Fig. 2.

The mobile node announces its information in the foreign agents on the lowest level foreign agent (FA.1.2.1.L) using an agent solicitation message, then the foreign agent broadcasts an agent advertisement message. The mobile node registers with its foreign agent using a registration message, then this message is forwarded through the foreign agents in the upper levels to the foreign agent of the lowest common ancestor (FA.1.L). In addition, the foreign agent of the lowest common ancestor sends a registration reply message to lower foreign agents and the mobile node. And, all prior information stored between the lowest common ancestor and the old foreign agents on the lowest level are removed using a tear down message. When the foreign agent on the upper level is the group leader of a lower level, since the two foreign agents are the same, only one register is transferred. In Fig. 2, we can see that registration request and reply message are exchanged between FA1.L and FA1.2.1.L in logical hierarchy architecture, yet in reality these messages are sent between FA1.2.1.L and FA1.1.2.L.

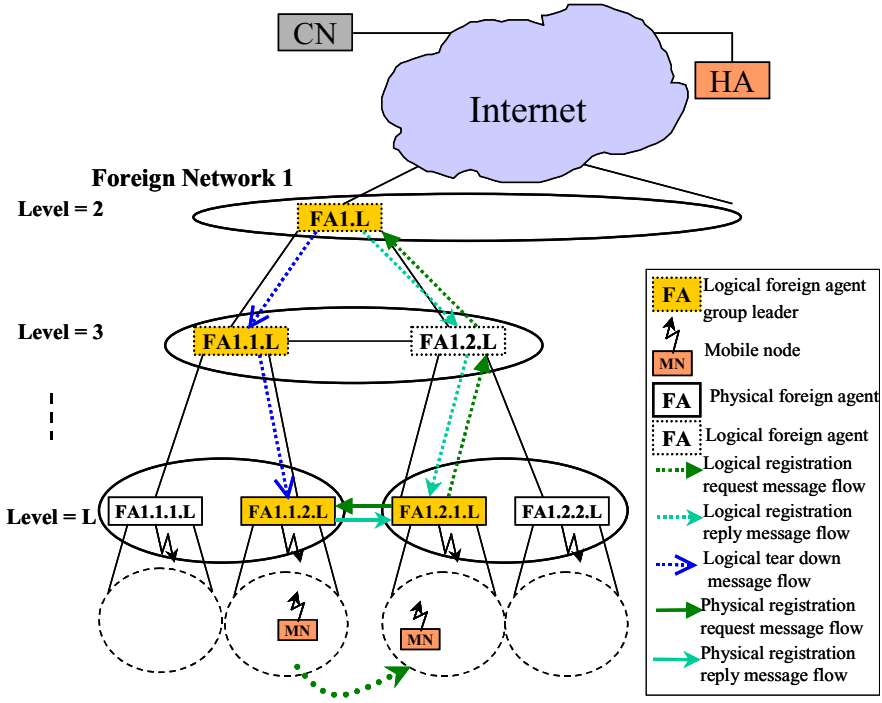


Fig. 2. Registration procedure in logical architecture

4 Performance Analysis

This section analyzes the proposed logical architecture described in section 2 along with the existing physical architecture, and then compares the costs. The comparison measures include the number of databases, registration update cost. For analysis purposes, a simplified hierarchical mobile computing environment was used. The notation used in the analysis is listed in Table 1 [10-11].

Registration requires databases, referred to as mobility agents, for storing the location information and profiles of the mobile nodes. The number of databases required is an important factor in the network construction, and in registration architecture, this quantity depends on the size of the network and depth of the layers.

The total number of the databases used in the physical hierarchy architecture is shown in equation (1). Here g means the number of databases in the foreign agent group, plus this architecture assumes that the g value is the same for all layers.

$$N_{PHY} = \sum_{i=1}^{L} (g^{L-i}) \tag{1}$$

Table 1. Notation for performance evaluation

Symbol	Meaning
L	Number of lowest level in network
S	Scope indication of neighborhood for registration update propagations
R_k	Cost of updating or querying databases up to level k from mobile node in physical layer architecture
A_k	Cost of updating or querying databases up to level k from mobile node in logical layer architecture
a_{ij}	Ancestors-are-siblings level of two nodes i and j
c_i	Cost of updating or querying level i mobility agent, $c_i = 0$, $i > L$
h, v, o, n, c	Subscripts used to represent home, visiting, old, and new locations of mobile node, and calling party, respectively
R	Cost of sending long distance message

In the proposed logical hierarchy architecture, logical nodes in an upper level represent the database of the foreign agent group leader in a lower level. Accordingly, while the physical hierarchy architecture includes a number of databases in each layer, the proposed logical hierarchy architecture only includes the databases from the lowest layer. The total number of databases in the proposed logical hierarchy architecture is shown in equation (2).

$$N_{LOG} = g^{L-1} \tag{2}$$

The costs of the registration update were analyzed for the existing physical hierarchy architecture and the proposed logical hierarchy architecture. Equation 3 presents the average registration update cost per move incurred by the Mobile IP scheme for the existing physical hierarchy architecture.

$$\begin{aligned} \overline{M}_{PHY} &= \sum_{i=S}^{L} P[a_{on} = i](R_{a_{on}} + R_{a_{on}+1} + 1) \\ &\quad + \sum_{i=1}^{S-1} P[a_{on} = i] \\ &\quad \cdot \left\{ \sum_{j=S}^{L} P[a_{hn} = j](2R_S + r + 1) + \sum_{j=1}^{S-1} P[a_{hn} = j](2R_S + 2r) \right\} \end{aligned} \tag{3}$$

The cost of updating/querying foreign agents in physical hierarchy architecture from the level-L agent to the level-k agent is given by

$$R_k = \sum_{i=k}^{L} c_i \tag{4}$$

where the cost is defined in Table 1.

Registration update costs in logical hierarchy architecture incurred by the Mobile IP scheme are shown in Table 2. Registration update in logical hierarchy architecture essentially requires updating pointers at foreign agents (see Fig. 2). Equation 5 shows the average registration update cost for proposed logical hierarchy architecture.

$$\begin{aligned} \overline{M}_{LOG} &= \sum_{i=S}^{L} P[a_{on} = i](A_{a_{on}} + A_{a_{on}+1} + 1) \\ &\quad + \sum_{i=1}^{S-1} P[a_{on} = i] \\ &\quad \cdot \left\{ \sum_{j=S}^{L} P[a_{hn} = j](2A_S + r + 1) + \sum_{j=1}^{S-1} P[a_{hn} = j](2A_S + 2r) \right\} \end{aligned} \tag{5}$$

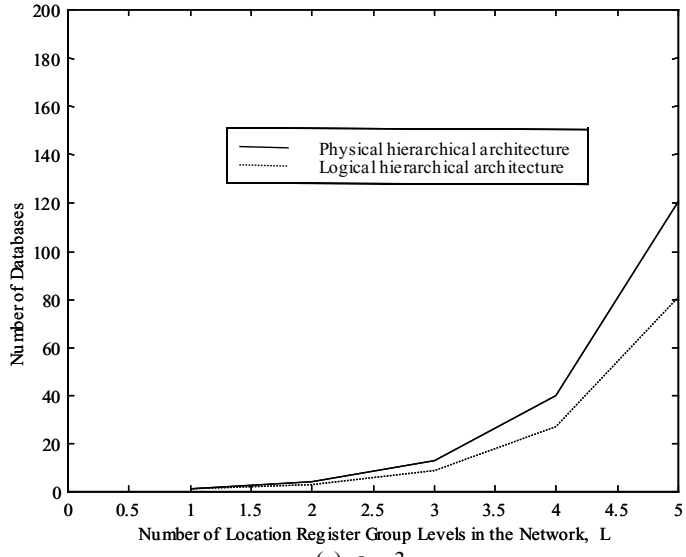
The cost of updating/querying foreign agents in logical hierarchy architecture from the level-L agent to the level-k agent is given by

$$A_k = \sum_{i=k}^{L-1} (1 - \frac{1}{g}) c_i + c_L \tag{6}$$

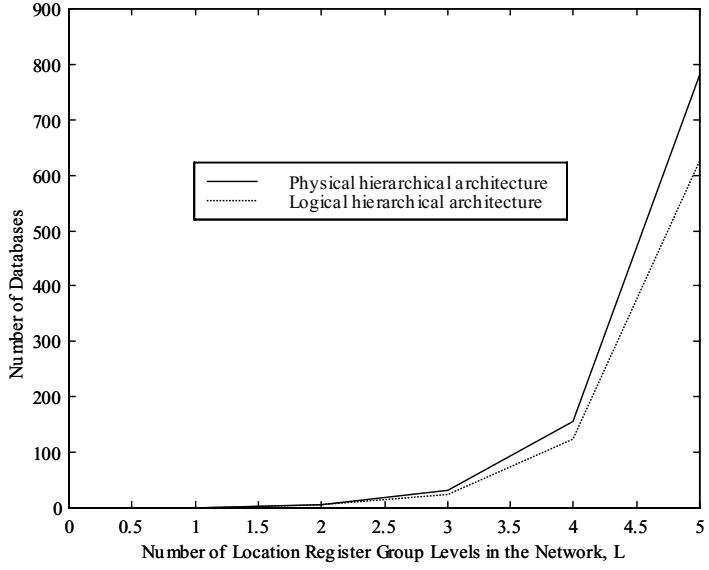
where the cost is defined in Table 1.

Table 2. Registration update cost in logical hierarchy architecture

Procedure	Relative locations	Cost
Power on	$a_{hv} \geq S$	$A_S + 1$
	$a_{hv} < S$	$A_S + r$
Move	$a_{on} \geq S$	$A_{a_{on}} + A_{a_{on}+1} + 1$
	$a_{on} < S$, $a_{hn} \geq S$	$2A_S + r + 1$
	$a_{on} < S$, $a_{hn} < S$	$2A_S + 2r$
Power off	$a_{hv} \geq S$	$A_S + 1$
	$a_{hv} < S$	$A_S + r$



(a) $g = 3$



(b) $g = 5$

Fig. 3. Comparison of the total databases according to the number of the layer levels

We present the total number of databases to evaluate the quantitative results for the existing physical architecture and proposed logical architecture for a mobile computing environment. It was assumed that the number of databases in the foreign agent group g in each layer was the same.

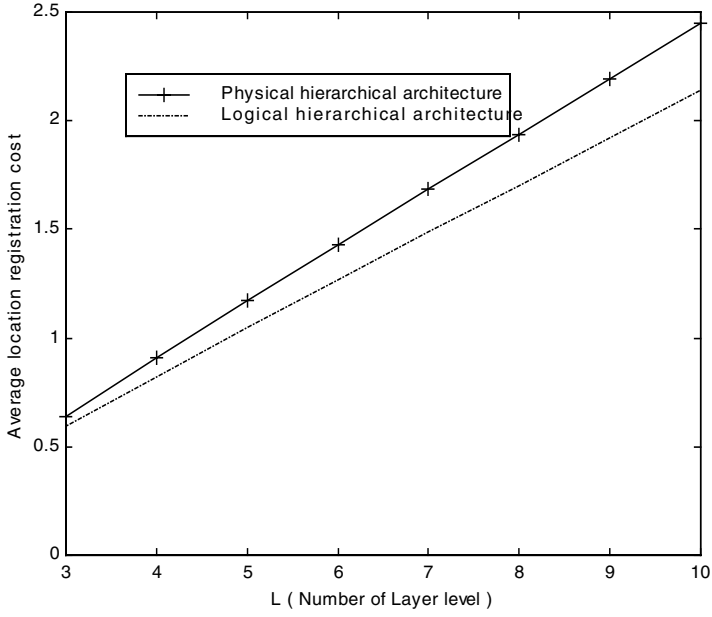


Fig. 4. Comparison of the average registration update costs between the existing physical architecture and the proposed logical architecture

Table 3. Input parameters

Parameters	Value
c_i, $1 \le i \le L$	1
g	3
r	5

The total number of databases required for each architecture was computed using equations (1) and (2) and the input values from Table 2. This process is shown in Fig. 3, which plots the total number of databases relative to the number of layers L. The results show that proposed logical architecture performed better than the existing physical architecture according to the number of layers.

This section presents the quantitative results of the comparison using equation (3) and (5). The values for the input parameters assumed in the numerical computation are shown in Table 3 [11].

The average registration cost of each architecture was computed using equation (3) and (5) and the input values from Table 3. This process is shown in Fig. 4, which plots the average registration update cost. The proposed logical architecture produced lower costs than the existing physical architecture. In fact, the proposed logi-

cal architecture performed better than the existing logical architecture at all layer values. It is interesting to note that the registration update cost of the proposed logical architecture was less than that of the existing physical architecture as the number of layers increased. The reason is the proposed logical architecture facilitates the situation where the upper register becomes the same as the lower register as the number of layers increases.

5 Conclusions

This paper presents a logical hierarchy architecture for mobility agents in mobile computing environments. The proposed logical hierarchy architecture of foreign agents organizes the lowest level physical foreign agents into clusters, called logical groups, which are then represented in the next level up by logical nodes. Thereafter, these logical nodes are regrouped into clusters and treated as single nodes in the next level up for all successive levels until all foreign agents are included in a tree-type logical hierarchy architecture. The analysis results demonstrate that the proposed logical hierarchy architecture could reduce the number of foreign agents along with the average registration update cost at the layer level.

References

1. F. Akyildiz, J. Mcnair, J. Ho, H. Uzunalioglu, and W. Wang, “Mobility Management for Next Generation Wireless Systems,” IEEE Proc. J., vol. 87, no. 8, pp. 1347-1384. Aug. 1999.
2. Gihwan Cho and L. F. Marshall, “ An Efficient Location and Routing Scheme for Mobile Computing Environments,” IEEE J. Select. Areas Comm., vol 13, no 5, June 1995.
3. A. Acharya, J. Li, B. Rajagopalan, and D. Raychaudhuri, “Mobility Management in Mobile Computing Environments,” IEEE Comm. Mag., vol. 35, no. 11, pp. 50-68, Nov. 1997.
4. M. Veeraraghavan, “A Distributed Control Strategy for Mobile Computing Environments,” ACM/Baltzer Wireless Networks J., pp. 323-339, 1995.
5. C. Perkins. IPv4 Mobility Support. IETF RFC 2002, October 1996.
6. C. Perkins. Mobile IP Design Principles and Practices. Number ISBN: 0-201-63469. Addison-Wesley Longman, Reading, MA, USA, 1998.
7. P. Calhoun and C. Perkins. Mobile IP Network Access Identifier Extension for IPv4. IETF RFC 2794, March 2000.
8. C. Perkins and K. Y. Wang, “Optimized smooth handoffs in Mobile IP," in Proceedings of the 4th IEEE Symposium on Computers and Communications, 1999.
9. D. Forsberg, J.T. Malinen, T. Weckstroem, and M. Tiusanen. Distributing Mobility Agents Hierarchically under Frequent Location Update. In Proc. of Sixth IEEE International orkshop on Mobile Multimedia Communications (MOMUC’99), San Diego, CA, USA, 1999.
10. M. Veeraraghavan, M.Karol, and K.Eng, “ Mobility and Connection Management in a Mobile ATM LAN,” IEEE J. Select. Areas Comm., vol.15, no. 1, pp.50-68, Jan. 1997.
11. M. Veeraraghavan and G. Dommety, “Mobile Location Management in Computing environmentss,” IEEE J. Select. Areas Comm., vol. 15, no. 8, pp. 1437-1454, Oct. 1997.

An Adaptive Security Model for Heterogeneous Networks Using MAUT and Simple Heuristics

Jongwoo Chae [1], Ghita Kouadri Mostéfaoui [2], and Mokdong Chung[1]

[1] Dept. of Computer Engineering, Pukyong National University,
599-1 Daeyeon-3Dong, Nam-Gu, Busan, Korea
madcow5980@hanmail.net, mdchung@pknu.ac.kr

[2] Software Engineering Group, University of Fribourg
Rue Faucigny 2, CH-1700 Fribourg, Switzerland
Ghita.KouadriMostefaoui@unifr.ch

Abstract. In this paper, we present an adaptive security model that aims at securing resources in heterogeneous networks. Traditional security model usually work according to a static decision-making approach. However, we may establish a better approach to heterogeneous networks if we use a dynamic approach to construct the security level. Security management relies on a set of contextual information collected from the user and the resource environments, and that infers the security level to enforce. These security levels are dynamically deduced using one of these two algorithms: MAUT and Simple Heuristics.

1 Introduction

Traditional security model usually work according to a static decision-making approach. However, we may establish a better approach to heterogeneous networks if we use a dynamic approach to construct the security level. In the current computing environment, heterogeneous networks are widely available and they have many different properties such as transmission speed, communication media, connectivity, bandwidth, range and etc. Moreover, many types of computing devices are widely used and they have diverse capabilities. To secure this diverse environment, we should adapt several security levels dynamically according to the diverse networks and computing devices. Unfortunately, these characteristics of heterogeneous networks and diverse computing capabilities are dynamically changing by contexts. To cope with this dynamic computing environment, we should make security level more adaptive.

In this paper, we develop an adaptive security model that dynamically adapts the security level according to a set of contextual information such as terminal types, service types, network types, user's preferences, information sensitivity, user's role, location, and time, using MAUT (Multi-Attribute Utility Theory) and Simple Heuristics in order to support secure transactions in the heterogeneous network.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 983–993, 2004.
© Springer-Verlag Berlin Heidelberg 2004

The remainder of the paper is organized as follows: Section 2 discusses related work. Section 3 focuses on the architecture of our system and the theoretical foundations of the used algorithms. Section 4 shows a case study, and Section 5 concludes this paper.

2 Related Work

From the large panel of definitions of "context," we retain Dey's definition [7]. Context may include physical parameters (type of network, physical location, temperature, etc), human factors (user's preferences, social environment, user's task, etc), and is primarily used to customize a system behavior according the situation of use and/or users' preferences. Context-aware computing is an emerging research field that concerns many areas as information retrieval [22], artificial intelligence [2], computer vision [20], and pervasive computing [8][15][23]. As a consequence many context-aware projects emerged as Cyberguide [11] and SART [1][16]. Even if the list is not exhaustive, most context-aware applications show a strong focus on location [17]. This situation is probably due to the lack of variety of sensors used and the difficulty in sensing high-level contexts, such as in a meeting or intrusion detection.

Considering context in security is a recent research direction. Most of the efforts are directed towards securing context-aware applications. In [4] and [6], Covington et al. explore new access control models and security policies to secure both information and resources in an intelligent home-environment. Their framework makes use of environment roles [5]. In the same direction, Masone designed and implemented RDL (Role-Definition Language), a simple programming language to describe roles in terms of context information [13]. There have also been similar initiatives in [19] and [14]. Interestingly, we observed that all previous work on combining security and context-aware computing follow the same pattern: using contextual information to enrich the access control model in order to secure context-aware applications with a focus on specific applications. The second main observation is that security decisions follow an old-fashioned rule-based formalism which does not consider systems and networks dynamics.

In a recent work [10], Kouadri and Brézillon propose a conceptual model for security management in pervasive environments. The proposed architecture aims at providing context-based authorizations in a federation of heterogeneous resources, and their architecture allows easy plugging of various formalisms for modeling context-based security policies. Our study is inspired from their model with a concrete implementation of the module that manages the context-based security policy.

Kouadri and Brézillon made the first attempt to define a security context [10]. Parts of contextual information are gathered from the pervasive environment where the other part is provided by the requesting user such as its preferences for instance.

Chung and Honavar [3] suggested the negotiating technique across multiple terms of transaction. Determining of security levels may resort to negotiating across multiple terms of transaction, such as terminal types, service types, user's preference, and the level of sensitivity of information.

Section 3 below describes in more detail the types of contextual information our system relies on, the theoretical foundations of MAUT and Simple Heuristics, which deduct more appropriate security level and how this security level will be applied.

3 A Context-Based Adaptive Security Model

This Section describes the physical architecture of our context-based adaptive security model. We consider a set of heterogeneous resources (typically printers, e-learning systems, and computing services) logically grouped into a federation; a university local area network for instance. Figure 1 illustrates the role of the context engine in mediating clients' requests to the protected resources available on the network federation. Users of these resources interact with the services using various means as home computers, handled devices and notebooks with wireless connection. The role of the context engine is to apply the context-based policy by implementing the logic that derives the security level to enforce between a service and its client in each situation (security context). Formally, in our system, a security context is a collection of context elements that are sufficient for applying a given security level. Context elements are predicates involving a number of terms that refer to the user, and computational security context. They include terminal types, service types, user's preference, and the level of sensitivity of information.

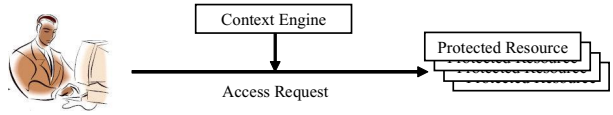


Fig. 1. Overall Architecture

3.1 Contextual Information for the Adaptive Security Model

In order to develop an adaptive protocol which could be used in the diverse environments, we need to define the security level explicitly. The development of such a protocol leads to overcome the limitation of the traditional security model, which sticks to a uniform use of cryptographic techniques, by introducing classification of security level according to domain dependent and independent aspects. Generally, we can classify the security contexts which affect the security level by the two aspects such as domain-independent and domain-dependent contexts.

A mathematical model for the adaptive security system based on contextual information is defined as follows:

$$
\begin{aligned}
&U = \sum_{i=1}^{n} k_i u_i(x_i),\ (0 \le U \le 1) \\
&SL = \lceil U * 10 \rceil / 2,\ (SL = 0, 1, \ldots, 5) \\
&P_{SL} = \{p_0, p_1, \ldots, p_5\} \\
&p_i = \{S_j, A_l, R_m\}
\end{aligned} \tag{1}
$$

This model determines the adaptive security level to meet the dynamic changes of environmental attributes in the heterogeneous networks. Based on this security level, this model adaptively adjusts the values of the environmental attributes of a security system such as algorithm type, key size, authentication method, and/or protocol type.

p_i consists of a tuple of (S_j, A_l, R_m), determined by the security policy, where S_j is encryption algorithm type, A_l is authentication type, and R_m is protocol type. U is the total utility value, u_i is a utility value of environmental attributes in the heterogeneous networks, and k_i is a scaling constant of the environmental attributes. SL represents security level from 0 through 5, where value 0 means we can not utilize the security system. The larger the number is, the stronger the strength of security is. Table 1 shows several algorithm types, and Table 2 illustrates protocol types and authentication methods.

Table 1. Algorithm types (S_j)

S_j	*Symmetric (Key size)*	*Asymmetric (Key size)*	*MAC*
S_0	DES	RSA(512)	MD5
S_1	3DES	RSA(512)	MD5
S_2	3DES	RSA(768)	SHA
S_3	AES(128)	RSA(1024)	SHA
S_4	AES(192)	RSA(1024)	SHA

Table 2. Protocol types (R_m) and Authentication methods (A_l)

R_m	*Protocol types*	A_l	*Authentication methods*
R_0	SPKI	A_0	Password based only
R_1	Wireless PKI	A_1	Certificate based
R_2	PKI	A_2	Biometric based
		A_3	Hybrid methods

3.2 Multi-attribute Utility Theory

Multi-Attribute Utility Theory is a systematic method that identifies and analyzes multiple variables in order to provide a common basis for arriving at a decision. As a decision making tool to predict security levels depending on the security context (network state, the resource's and user's environments, etc), MAUT suggests how a decision maker should think systematically about identifying and structuring objectives, about vexing value tradeoffs, and about balancing various risks. The decision maker assigns utility values to consequences associated with the paths through the decision tree. This measurement not only reflects the decision maker's ordinal rank-

ings for different consequences, but also indicates him relative preferences for lotteries over these consequences [9].

According to MAUT, the overall evaluation $v(x)$ of an object x is defined as a weighted addition of its evaluation with respect to its relevant value dimensions [21][1]. The common denominator of all these dimensions is the *utility* for the evaluator [18]. The utility quantifies the personal degree of satisfaction of an outcome.

The MAUT algorithm allows us to maximize the expected utility in order to become the appropriate criterion for the decision maker's optimal action.

3.3 Simple Heuristics

The Center for Adaptive Behavior and Cognition is an interdisciplinary research group founded in 1995 to study the psychology of bounded rationality and how good decisions can be made in an uncertain world. This group studies Simple Heuristics. The first reason why we use Simple Heuristics is that security level can be decided without user's detailed preference. And the second reason is that it is difficult to predict the preferences of users concerning the attributes of the security level. It is also difficult for even users to determine their preferences quantitatively which is related to the attributes of the security level.

By the way, different environments can have different specific heuristics. But specificity can also be a danger if a different heuristic were required for every slightly different environment, we would need an unworkable multitude of heuristics. Fast and frugal heuristics avoid this trap by their simplicity and enable them to generalize well to new situations. One of fast and frugal heuristics is Take The Best which tries cues in order, searching for a cue that discriminates between the two objects. It serves as the basis for an inference, and all other cues are ignored. Take The Best outperforms multiple regression, especially when the training set is small [12].

3.4 Security Policy Algorithm

We begin by presenting the security policy algorithm that dynamically adapts the security level according to the domain independent properties such as terminal types, and the domain dependent properties such as the sensitivity of information using MAUT and Simple Heuristics. The variables of the algorithms are as follow:

1. domain dependent variables $I = (i_1, i_2, ..., i_n)$: data size, computing power, network type, terminal type, and so on.
2. domain independent variables $X = (x_1, x_2, ..., x_n)$: user attributes, system attributes
3. security level $SL = (0, 1, 2, ..., 5)$: The larger the number is, the stronger the strength is. If SL is 0, we can not utilize the security system.

[1] [21] describes other possibilities for aggregation.

The overall algorithms for determining adaptive security level are as follows.

```
SecurityLevel(securityProblem)
// securityProblem: Determining security level

// Utilization of domain independent properties
   calculate SL by I end
   if SL = 0 then return SL // no use of security
system

// Utilization of domain dependent properties
// user select a strategy between MAUT and S. Heuris-
tics
   if MAUT then SL = MAUT(X)
   if Simple Heuristics then SL = TakeTheBest(X);
   return SL; end;
```

```
MAUT(X)
// Determine total utility function by the interaction
// with  the user according to MAUT

   u(x1,x2,…,xn)=k1u1(x1)+k2u2(x2)+… +knun(xn);
// ki is a set of scaling constants
// xi is a domain dependent variable, where ui(x°i)=0,
// ui(x*i)=1,and ki is positive scaling constant for
all i

   ask the user's preference and decide ki
   for i = 1 to n
     do ui(xi) = GetUtilFunction(xi);
   end
   return u(x1,x2,…xn); end;
```

```
GetUtilFunction(xi)
// Determine utility function due to users' prefer-
ences
// xi is one of domain dependent variables

   uRiskProne : user is risk prone for xi // convex
   uRiskNeutral : user is risk neutral for xi // lin-
ear
   uRiskAverse : user is risk averse for xi //concave
   x : arbitrary chosen from xi
   h : arbitrary chosen amount
   <x+h, x-h> : lottery from x+h to x-h

// where the lottery (x*, p, x°) yields a p chance at
x*   // and a (1-p) chance at x°
   ask user to prefer <x+h, x-h> or x; // interaction
   if user prefer <x+h, x-h> then
     return uRiskProne;              // e.g. u = b(2^cx-
1)
   elseif user prefer x then
     return uRiskAverse;             // e.g. u =
blog2(x+1)
   else return uRiskNeutral;  end; // e.g. u = bx
```

```
TakeTheBest(u(x1, x2, …, xn))
// Take the best, ignore the rest
   u(x1, x2, . . , xn) : user's basic preferences
// if the most important preference is xi, then only
xi      // is considered to calculate SL
// The other properties except xi are ignored

   u(x1, x2, …, xn) is calculated by only considering xi
   SL is calculated by the value of u(x1, x2, …, xn)
   return SL; end;
```

4 A Case Study

In Section 3, we discussed the mathematical foundations for both the MAUT and Simple Heuristics. In this Section, we present a concrete example that makes use of our security management system and that relies on the set of contextual information described in Section 3.

4.1 An Example of Determining a Utility Function in MAUT

For instance, if the utility function $u(x_1, x_2, x_3)$ with three attributes is additive and utility independent, then

$$U(x_1, x_2, x_3) = k_1u_1(x_1) + k_2u_2(x_2) + k_3u_3(x_3), \text{ where } u_i(x^o_i) = 0, u_i(x^*_i) = 1, \text{ for all } i \quad (2)$$

where the least preferred consequence, $u_i(x^o_i) = 0$, the most preferred consequence, $u_i(x^o_i) = 1$ for all i.

And then, we ask the decision maker some meaningful qualitative questions about k_i's to get some feeling for their values. For instance, "Would you rather have attribute $\boldsymbol{X}_1$ pushed to x^*_1 than both attributes $\boldsymbol{X}_2$ and $\boldsymbol{X}_3$ pushed to x^*_2 and x^*_3?" A yes answer would imply $k_1 > k_2 + k_3$, which means $k_1 > .5$. We then ask "Would you rather have attribute $\boldsymbol{X}_2$ pushed from x^o_2 to x^*_2 than $\boldsymbol{X}_3$ pushed from x^o_3 to x^*_3?" A yes answer means $k_2 > k_3$. Suppose that we assess $k_1 = .6$, that is, the decision maker is indifferent between (x^*_1, x^o_2, x^o_3) and the lottery $<(x^*_1, x^*_2, x^*_3), .6, (x^o_1, x^o_2, x^o_3)>$, where the lottery (x^*, p, x^o) yields a p chance at x^* and a $(1- p)$ chance at x^o. Then $(k_2 + k_3) = .4$ and we ask "What is the value of p so that you are indifferent between (x^o_1, x^*_2, x^o_3) and $<(x^o_1, x^*_2, x^*_3), p, (x^o_1, x^o_2, x^o_3)>$?" If the decision maker's response is .7, we have $k_2 = p(k_2 + k_3) = .28$

Then, $u(x_1, x_2, x_3) = .6u_1(x_1) + .28u_2(x_2) + .12u_3(x_3)$. Each $u_i(x_i)$ function is determined by the interaction with the user as follows: If a decision maker is *risk prone* then $u_i(x_i)$ is convex function, such as $b(2^{cx} -1)$; else if a decision maker is *risk averse* then $u_i(x_i)$ is concave function, such as $blog_2(x+1)$; else if a decision maker is *risk neutral* then $u_i(x_i)$ is linear function, such as bx; where b, c > 0 constants.

4.2 An Example of Determining Security Policy and Access Policy

Table 3 is a typical example of security policy. In Table 3, x_{att} is the strength of the cipher, x_{auth} is the authentication method, and x_{res} is the level of protection of the resource to which the user is trying to access. The unit of x_{att} is MIPS-Years which is a metric of time to need to break the protected system. *comp* is computing power for message encryption/decryption, *nType* is network type, and *tType* is terminal type, respectively. We need to have a terminal equipped with better than 200 MHz CPU and bandwidth over 100 Kbps to access to the protected resource A. Also we can use PC, PDA, or Cellular phone. User's preference determines the shape of the utility function as discussed in *GetUtilFunction()*, subsection 3.4.

Security policy determines the environmental attributes which will be used in the adaptive security level algorithm, constructs the utility function according to the user's preference, and finally determines the security level by using security level algorithm, *SecurityLevel()*. Access policy provides access right or denial to the protected resource according to the security level and user's privilege.

Table 3. An example of security policy

A Security Policy for *Protected Resource A*	
Action	reading
Utility Function	$u(x_{att}, x_{auth}, x_{res}) = k_{att}\, u(x_{att}) + k_{auth}\, u(x_{auth}) + k_{res}\, u(x_{res})$;
Security Contexts	*comp* $\geq$ 200 MHz; *nType* $\geq$100 Kbps; *tType* = PC/PDA/Cell;
User's Preference	*uRiskProne* = $2^{2(x-1)}$; *uRiskNeutral* = x; *uRiskAverse* = $\log_2(x+1)$;

Table 4 is conversion table for environmental attributes whose utility value is mapped from 0 through 1. Each value may be used to calculate the total utility function value.

Table 4. Conversion table for environmental attributes

utility value / utility attribute	0.2	0.5	0.8	1.0
x_{att} (MIPS-Years)	$\geq 10^{0.5}$	$\geq 10^{3}$	$\geq 10^{7}$	$\geq 10^{11}$
x_{auth} (Authentication)	Password only	Certificate	Biometric	Hybrid
x_{res} (level of protection)	No	Low	Medium	High

Table 5 is an example of access policy where reading or writing access right is given to the user according to the security level, user's role, and/or time attributes. *SL* is the lower bound of security level. Any user cannot adopt *SL* lower than 3 for write operation. If the user is administrator and *SL* is higher than 3, then he or she can write.

Table 5. An example of access policy

An Access Policy for Protected Resource A
If ((SL 2) **and** ((Role = administrator) **or** ((Role = user) **and** (Date = Weekdays **and** 8:00 < Time < 18:00)))) **then** resource A can be read
If ((SL 3) **and** (Role = administrator)) **Then** resource A can be written

4.3 The Strengths of the Proposed Model

The strengths of the proposed model are as follows: Firstly, traditional security model usually work according to a static decision-making approach since the same authenti cation and authorization protocol might be used to the diverse protected resources, for instance. This might result in the waste of system resources, such as excessive usage of CPU and excessively high network bandwidth. In the proposed model, we can reduce the waste of resources by adaptively applying appropriate cryptographic techniques and protocols according to the characteristics of the resources. Therefore, the proposed model increases efficiency and availability of the resources.

Secondly, in terms of system protection, our model is more secure than traditional one. When the system identifies possibilities of attacks or vulnerabilities of the resources, our model protects the system by adaptively decreasing security level of the resource. When the security level is decreased, the access request might be denied by applying the rule sets of the access policy.

Finally, the traditional security systems can not consider user's security preference. In the contrast, our model can reflect the user's preference. Therefore, the result of the same access request could be quite different although any other contexts are the same.

5 Conclusion and Future Work

In this paper, we presented an adaptive security model that provides adaptive security policies for heterogeneous networks. Adaptability is expressed using a set of contextual information about all the parties included in the interaction, namely, the protected resource, the requesting user, and the network which represents the working platform for the interaction. For each security context, a security level is enforced by the mean of two algorithms; MAUT and Simple Heuristics. Our system has been applied to a university local area network with a set of heterogeneous services as printer services, e-learning systems, etc. Moreover, the proposed architecture could be applied to any network that offers different types of services and resources, in order to provide context-based fine-grained access to these resources.

In the future, we will analyze quantitatively the effectiveness of the proposed adaptive security model through a simulation or a real implementation in the heterogeneous networks.

References

[1] P. Brézillon, et al., "SART: An intelligent assistant for subway control," *Pesquisa Operacional, Brazilian Operations Research Society*, vol. 20, no. 2, 2002, pp. 247-268.

[2] P. Brézillon, "Context in Artificial Intelligence: I. A survey of the literature," *Computer & Artificial Intelligence*, vol. 18, no. 4, 1999, pp. 321-340, http://www-poleia.lip6.fr/~brezil /Pages2/Publications/CAI1-99.pdf.

[3] M. Chung and V. Honavar, "A Negotiation Model in Agent-Mediated Electronic Commerce," *Proc. IEEE Int'l Symposium on Multimedia Software Engineering*, Taipei, Dec. 2000, pp. 403-410.

[4] M.J. Covington, et al., *A Security Architecture for Context-Aware Applications*, tech. report, GIT-CC-01-12, College of Computing, Georgia Institute of Technology, May 2001.

[5] M.J. Covington, at el., "Securing Context-Aware Applications Using Environment Roles," *Proc. 6th ACM Symposium on Access Control Models and Technologies*, Chantilly, VI, USA, May, 2001, pp. 10-20.

[6] M.J. Covington, at el., "A Context-Aware Security Architecture for Emerging Applications," *Proc. Annual Computer Security Applications Conf. (ACSAC)*, Las Vegas, Nevada, USA, Dec. 2002.

[7] A.K. Dey, Ph. D. dissertation, *Providing Architectural Support for Building Context-Aware Applications*, Georgia Institute of Technology, 2000.

[8] K. Henricksen, at el., "Modeling Context Information in Pervasive Computing Systems," *Proc. 1st Int'l Conf., Pervasive 2002*, Zurich, Springer Verlag, Lecture Notes in Computer Science, vol. 2414, 2002, pp. 167-180.

[9] R.L. Keeney and H. Raiffa, *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*, John Wiley & Sons, New York, NY, 1976.

[10] G. K. Mostéfaoui and P. Brézillon, "A generic framework for context-based distributed authorizations," *Proc. 4th Int'l and Interdisciplinary Conf. on Modeling and Using Context (Context'03)*. LNAI 2680, Springer Verlag, pp. 204-217.

[11] S. Long, et al., "Rapid prototyping of mobile context-aware applications: The cyberguide case study," *Proc. the 1996 Conf. Human Factors in Computing Systems (CHI'96),* 1996, pp. 293-294.

[12] L. Martignon and U. Hoffrage, *Why Does One-Reason Decision Making Work? In Simple Heuristics That Make Us Smart*, Oxford University Press, New York, 1999, pp. 119-140.

[13] C. Masone, *Role Definition Language (RDL): A Language to Describe Context-Aware Roles*, tech. report, TR2001-426, Dept. of Computer Science, Dartmouth College, May 2002.

[14] P. Osbakk, and N. Ryan, "Context Privacy, CC/PP, and P3P," *Proc. UBICOMP2002 - Workshop on Security in Ubiquitous Computing*, 2002, pp. 9-10.

[15] A. Rakotonirainy, *Context-oriented programming for pervasive systems*, tech. report, University of Queensland, Sep. 2002.

[16] *The SART Project*, http://www-poleia.lip6.fr/~brezil/SART/index.html.

[17] A. Schmidt, et al., "There is more to context than location," *Computers and Graphics*, vol. 23, no. 6, Dec. 1999, pp. 893-902.

[18] R. Schäfer, "Rules for Using Multi-Attribute Utility Theory for Estimating a User's Interests," *Proc. 9th GI-Workshop. ABIS-Adaptivität und Benutzermodellierung in interaktiven softwaresystemen*, Dortmund, Germany, 2001.

[19] N. Shankar, D. Balfanz, "Enabling Secure Ad-hoc Communication Using Context-Aware Security Services (Extended Abstract)," *Proc. UBICOMP2002 -Workshop on Security in Ubiquitous Computing*.

[20] T. M. Strat, et al., *Context-Based Vision. Chapter in RADIUS: Image Understanding for Intelligence Imagery*, O. Firschein and T.M. Strat, Eds., Morgan Kaufmann, 1997.

[21] D. Winterfeld, von and W. Edwards, *Decision Analysis and Behavioral Research*, Cambridge, England: Cambridge University Press, 1986.

[22] M. Weiser, "The computer for the 21st Century," *Scientific American*, vol. 265, no. 3, 1991, pp. 66-75.

[23] S. S. Yau, et al., "Reconfigurable Context-Sensitive Middleware for Pervasive Computing," *IEEE Pervasive Computing, joint special issue with IEEE Personal Communications*, vol. 1, no. 3, July-September 2002, pp. 33-40.

A Hybrid Restoration Scheme Based on Threshold Reaction Time in Optical Burst-Switched Networks

Hae-Joung Lee[1], Kyu-Yeop Song[2], Won-Ho So[3], Jing Zhang[4], Debasish Datta[5], Biswanath Mukherjee[4], and Young-Chon Kim[1]

[1] Dept. of Computer Engineering, Chonbuk National University, Jeonju 561-756, Korea
{lhj9238, yckim}@chonbuk.ac.kr
[2] Dept. of Info & Com Engineering, Chonbuk National University, Jeonju 561-756, Korea
[3] Dept. of Computer Education, Sunchon National University, Sunchon 561-756, Korea
[4] Dept. of Computer Science, University of California, Davis, CA 95616, USA
[5] Dept. of Electronics and Electrical Commu. Engineering, IIT, Kharagpur 721302, India

Abstract. Optical Burst-Switched (OBS) networks usually employ one-way reservation by sending a burst control packet (BCP) with a specific offset time, before transmitting each data burst frame (BDF). Therefore, a fiber link failure may lead to loss of several BDFs as the ingress nodes sending these BDFs remain unaware of the failed link till they receive the failure indication signal (FIS). In this paper, we propose a hybrid restoration scheme, wherein we employ a novel combination of sub-path as well as path restoration schemes. In particular, the upstream node preceding the failed link employs sub-path restoration as soon as it detects the failure. Thereafter, when the source ingress node receives the FIS message, it (source ingress node) takes over the responsibility and employs path restoration. However, sub-path restoration is exercised by the upstream node only if the remaining time intervals left for the yet-to-arrive BDFs (for which BCPs have already left the upstream node) are less than a minimum time interval (called as threshold reaction time). Performance of the proposed scheme is evaluated through extensive computer simulation using OPNET. Our results indicate that the proposed restoration scheme significantly reduces burst losses due to link failure and, thus, outperforms the existing restoration schemes.

1 Introduction

The rapid growth in demand for IP traffic has led to a considerable shift in telecommunication trends from Voice-centric to IP-centric networks, and many multimedia applications require high-speed data transfers and transmission channels offering high reliability. Given this situation, wavelength-division multiplexing (WDM) technologies with enormous bandwidth capacity are expected to play a dominant role in such networks. WDM technologies allow optical fibers to support diverse traffic demands of the next-generation networks and multiple communication channels concurrently.

A. Laganà et al. (Eds.): ICCSA 2004, LNCS 3046, pp. 994–1003, 2004.
© Springer-Verlag Berlin Heidelberg 2004

As a consequence, the switching scheme in WDM networks becomes more complex than that in traditional networks.

Existing optical switching techniques can be broadly classified into three categories: optical circuit switching (OCS), optical packet switching (OPS), and optical burst switching (OBS). In recent years, OBS has received considerable research interest as a promising technology for the next-generation WDM networks [1-3]. The advantage of this technology is that it makes use of the past experience of OCS, while ensuring efficient bandwidth utilization on a fiber link, just as OPS does. In OBS networks, one wavelength (channel) on each link is reserved for control information. This separation of control and data channels simplifies the data path implementation, allowing for better use of optical switching technologies.

In OBS networks, a link failure can lead to considerable data loss and a significant reduction in Quality of Service (QoS). Moreover, the choice of offset time is one of the most critical issues, which can significantly influence the burst loss phenomenon in OBS networks. Therefore, in this paper, we propose a hybrid restoration scheme based on a combination of sub-path as well as path restoration scheme.

The rest of the paper is organized as follows. In Section 2, we briefly review the background of OBS networks. In Section 3, we provide an analysis of how burst loss is affected by offset time during link failure. In Section 4, we propose our restoration scheme based on a combination of sub-path as well as path restoration scheme. In Section 5, we present the result of computer simulations, conducted to evaluate the burst loss probability. Finally, Section 6 presents concluding remarks on our work.

2 OBS Networks and Recovery Schemes

Each data burst in an OBS network consists of a burst control packet (BCP) and a data burst frame (BDF). The information of the data burst length and offset time is carried in the BCP. The intrinsic feature of the OBS is the separation of transmission and switching between BCP and BDF. First, a BCP is sent out to set up a connection, and this is followed by the transmission of the corresponding BDF after a predefined offset time, without waiting for an acknowledgement for the establishment of the connection. When the BCP arrives at an intermediate node, it is converted into an electrical signal for switch reconfiguration and regeneration of BCP for onward reservation. Thus, the corresponding BDF can pass through the pre-configured optical switches without any O/E/O conversion.

At the ingress edge router, the offset time is predefined and calculated based on the total processing time needed by a BCP on its way to the destination egress router. As shown Fig. 1, a source (ingress) node sends out a BCP, which is followed by a burst after a base offset time, $t \geq \sum_{h=1}^{H} \delta(h)$, where $\delta(h)$ is the expected processing delay at hop $1 \leq h \leq H$ (Fig. 1, where $H = 3$ and $\delta(h) = \delta$). Because the BDF is buffered at the source in electronic domain during the offset time, fiber delay lines are not needed at any of the intermediate nodes, in order to delay the BDF while the BCP is being processed.

OBS network employ one-way reservation. Therefore, a fiber link failure may lead to loss of several BDFs as the ingress nodes sending these BDFs remain unaware of the failed link till they receive the failure indication signal. Depending on the timescales in which the spare capacity is allocated, there are essentially two techniques that can be used for the management of such failures: protection and restoration. Generally, protection is more expensive in terms of resources, since it requires that spare capacity be pre-allocated to establish backup paths, which can only be used in the case of failure, and remain unused until a fault occurs. However, in the case of restoration, more time may be required to reestablish the connection, because setting up a backup path in real time may involve dynamic route calculation and spare capacity allocation [4-8].

The total recovery time ranges from ≈50ms in protected SDH/SONET networks to more than 40s in IP networks with fault notification. Therefore, if it is possible to guarantee a 50ms recovery time in OBS networks, the restoration scheme would be the best option, because link or node faults occur rarely.

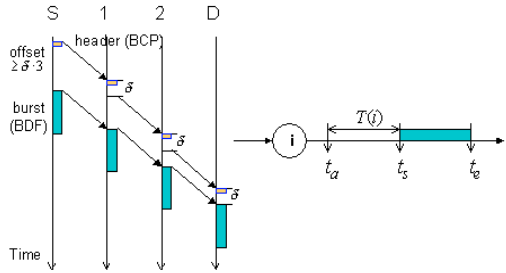


Fig. 1. Offset time

3 Burst Losses due to Link Failure in Existing Restoration Schemes: Analytical Model

In OBS networks, when a link failure occurs, not only can both BCPs and BDFs be lost due to the failed link, but also in some circumstances BDFs will be lost even if the corresponding BCPs already successfully passed through the failed link. This leads to serious problems, such as QoS, network throughput, utilization, etc.

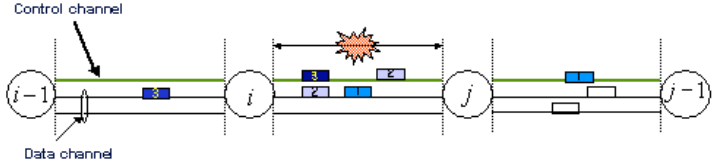


Fig. 2. The different cases of data burst loss due to link failure

In Fig. 2, we present various cases of data burst loss due to link failure, depending on the offset time. We assume that link failure occurs between nodes *i* and *j*.

Case 1. BCP 1 is passing through node *j*, but BDF 1 is located in the failed link. Although BDF 1 is lost due to link failure, BCP1 continues on its path to the destination in order to reserve the wavelength for the corresponding BDF 1 from node *j* to the destination. In this case, due to the offset time, two problems are encountered: loss of the BDF and waste of resources.

Case 2. In this case, there is a very small offset time between BCP 2 and BDF 2, and both of them are present in the failed link at the same time. Therefore, both BCP 2 and BDF 2 are lost. This case is different from Case 1 from the viewpoint of the resources involved, because no channel reservation is made.

Case 3. BCP 3 is in the failed link but has not yet arrived at node *i*. In this case, there are two possible scenarios, depending on the length of the offset time. In one of the two cases, if the offset time is long enough for a new BCP 3 to be generated, this replacement packet can use a backup path. In the other case, if the offset time is very short, BDF 3 will be lost.

In view of the above, we can classify the effect of offset time on burst loss into three different categories. In category 1 and 2, since the BDF is lost (with or without the BCP also being lost), it cannot be restored. In category 3, however, if the offset time is long enough for a new BCP to be regenerated, the burst will not be lost and, hence, we can still send the burst to its destination safely.

Let us assume that OBS nodes generate bursts following a Poisson distributed arrival rate with an average value of λ and an exponentially distributed service rate with an average value of $\mu = L/R$ with L as the BDF length and R as the per-channel transmission rate. The offered load of the OBS node is thus given by $\rho = \lambda/(\mu k) = r/k$, where $r = \lambda/\mu$, and the total offered load is $\rho = \sum_{i=0}^{n-1} \rho_i$. Note that, for simplicity, we assume that OBS nodes operate with bidirectional links, without any buffer and each link carries *k* wavelengths. So, the burst arrival rate λ can be expressed as

$$\lambda = \frac{2\rho\mu kE}{nH} \tag{1}$$

where,

E = Total number of links, n = Total number of nodes,

H = Average number of hops

Next, we define burst (i.e., a BDF) loss probability, which is determined using the Erlang's loss formula (M/M/k/k) as

$$B(\rho,k) = \frac{r^k/k!}{\sum_{m=0}^{k} r^m/m!} = P_{loss} \text{ (say)} \tag{2}$$

So, the probability for successful transmission of a BDF is $p_{succ} = (1 - p_{loss})$. Thus the probability of successful arrival of a BDF from its source to node *i* is $p_{succ} = (1 - p_{loss})^{h_i}$. Therefore, the offered loads of node *i* and node *j* can be written as

$$\rho_i = \rho(1 - p_{loss})^{h_i}, \ \rho_j = \rho(1 - p_{loss})^{h_j} \tag{3}$$

If the number of primary paths traversing link <*i, j*> is *N*, then the number of burst losses which occur in an OBS network just when link failure takes place is related to the arrival rate of *N* node pairs. The arrival rate at node *i*, λ_{ij}, can be written as

$$\lambda_{ij} = \sum_{s=1}^{N} \sum_{d=1}^{N} (\rho_s^d (1 - p_{loss})^{h_i^s} P_{ij}^{sd}) \quad s \neq d, i \neq j \tag{4}$$

Similarly the arrival rate at the upstream node *j*, λ_{ji}, is expressed as

$$\lambda_{ji} = \sum_{s=1}^{N} \sum_{d=1}^{N} (\rho_s^d (1 - p_{loss})^{h_j^s} P_{ji}^{sd}) \quad s \neq d, i \neq j \tag{5}$$

where,

h_{ij}^{sd} = Number of hops from source node to node *i,*

ρ_s^d = Offered load at the source node,

P_{ij}^{sd} = Probability that the primary path *(s,d)* passes through the failed link <*i, j*>

So, the number of bursts located in the failed link at the moment of failure is written as $Link_{BDF} = p_d(\lambda_{ij} + \lambda_{ji})$. The number of bursts lost during the interval of offset time is $Offset_{BDF} = t(\lambda_{ij} + \lambda_{ji})$. Therefore, the total number of bursts which are lost at the time of link failure, $Fault_{BDF}$, can be written as

$$Fault_{BDF} = Link_{BDF} + Offset_{BDF} = (p_d + t)(\lambda_{ij} + \lambda_{ji}) \tag{6}$$

Next, let us consider the restoration scheme in the case of link failure. In the *path restoration* scheme, the source node continues sending bursts to the upstream node until it has received a fault indication signal (FIS) message. The arrival rate of each node during the time that the upstream node requires to send an FIS message to the source node is given by

$$\lambda'_{ij} = \sum_{s=1}^{N} \sum_{d=1}^{N} \sum_{k=1}^{h_{ij}^{sd}} (\rho_s^d (1 - p_{loss})^{h_i^s} k P_{ij}^{sd}) \quad s \neq d, i \neq j \tag{7}$$

$$\lambda'_{ji} = \sum_{s=1}^{N} \sum_{d=1}^{N} \sum_{k=1}^{h_{ji}^{sd}} (\rho_s^d (1 - p_{loss})^{h_i^s} k P_{ji}^{sd}) \quad s \neq d, i \neq j \tag{8}$$

Therefore, the total number of bursts lost is expressed as

$$Fault_{BDF} = Link_{BDF} + Offset_{BDF} + FIS_{BDF} = (p_d + t)(\lambda_{ij} + \lambda_{ji}) + FIS_{BDF}(\lambda'_{ij} + \lambda'_{ji}) \tag{9}$$

In the *sub-path restoration* scheme, the total number of bursts lost is the same as that in the path restoration scheme, except that, in this case, failure detection takes place almost immediately at the upstream node. Therefore, the total number of bursts lost is given by

$$Fault_{BDF} = Link_{BDF} + Offset_{BDF} = (p_d + t)(\lambda_{ij} + \lambda_{ji}) \tag{10}$$

4 Hybrid Restoration Scheme Based on Threshold Reaction Time

The proposed scheme employs a novel combination of sub-path as well as path restoration schemes. It may be noted that, the upstream node immediately preceding the failed link comes to know about the failed link much earlier than the source ingress nodes whose BDFs get lost due to the link failure. Hence, the upstream node, having known about the link failure, takes up the initial step and employs a sub-path restoration. Thereafter, when the FIS message reaches the source node, latter (i.e., the source node) takes over the responsibility from the upstream node and employs path restoration. However, the sub-path restoration is exercised by the upstream node only if the remaining time intervals left for yet-to-arrive BDFs (whose BCPs have already passed through the upstream node – i.e., category 3 BDFs, as defined in Section 3) are less than a minimum time duration, called threshold reaction time. The threshold reaction time is expressed as

$$TRT = \textit{backup sub-path set up time} + \textit{time required to recreate the modified BCP}$$

So, when a link failure occurs, first our scheme handles the category 3 BDFs (i.e., the BDFs which are yet is arrive at the upstream node (where the link failure has been detected), corresponding BCPs have already left the upstream node) as follows. The SCU in the upstream node compares the threshold reaction time with the remaining time for yet-to-arrive category 3 BDFs. If the threshold reaction time is less than the time remaining before the arrival of the BDF, then the upstream node can find out an appropriate sub-path and accordingly recreate a BCP. Thereafter, the BDF can be sent to the destination node safely. Otherwise, the burst will be lost. Fig. 4 shows the proposed hybrid restoration scheme based on the threshold reaction time appropriate network.

As discussed above, the proposed scheme adopts a hybrid approach, wherein initial failure management is carried out by the upstream node (nearest to the failure location) using sub-path restoration based on threshold reaction time. In the mean time, FIS message can arrive the source node and thereafter the source node takes over the responsibility of failure management by using path restoration. The upstream node being nearer to the failure location can respond fast to the failure and hence takes care gracefully the initial crisis caused by the link failure. However, with the proposed restoration scheme, a BDF that arrives at the upstream node immediately following a link failure (or with a delay less than the threshold time) will get lost.

We examine the above scheme through computer simulation using OPNET. We also examine the existing schemes (sub-path and path restoration) by using OPNET as well as analytical model presented in Section 3. This helps us to validate the simulation methodology, which we subsequently extend to investigate the proposed hybrid restoration scheme based on threshold reaction time.

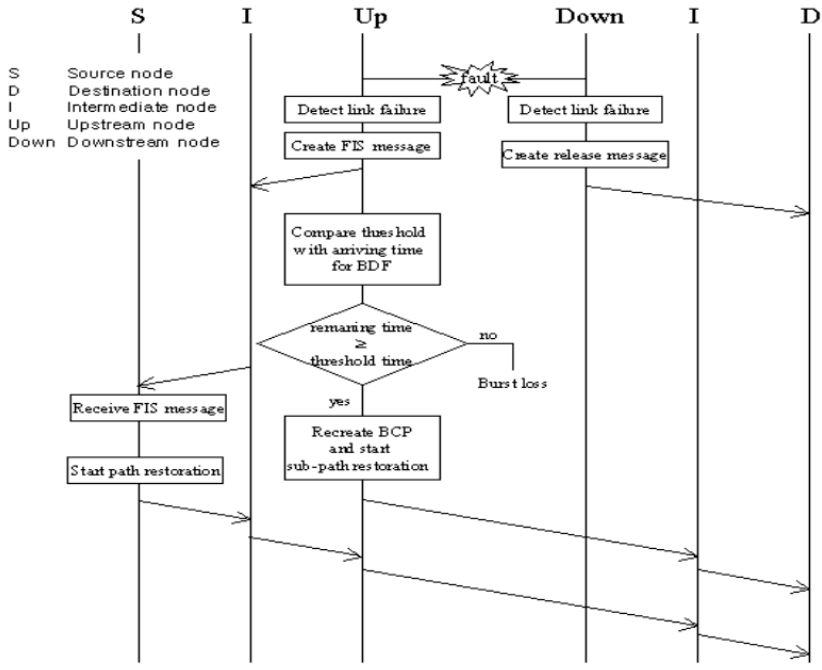


Fig. 4. The new restoration scheme

5 Numerical Results and Discussion

We consider the NSFNET network topology for performance evaluation. We assume that each node is capable of performing wavelength conversion. Every link consists of 9 wavelengths and is bi-directional, with one wavelength being used for BCPtransmission. The link transmission rate is 1Gbps. The fiber length is set to 1000km for each link, such that the propagation delay between the nodes is 1_{ms}.

The destination for each BDF is selected at random from a uniform distribution among the other nodes in the network. The shortest path between a pair of nodes is the hop-based distance. Each node generates BDFs, and the sizes of these BDFs are obtained form an exponential distribution, with the mean size being 100,000 bits. The simulation duration is 5 seconds and the start time of link failure is 2.1 seconds, wherein each case includes only one link failure (we select the link between node 7 to node 8).

Fig. 5 shows the plots of number of bursts lost vs. offered load for existing restoration schemes. Fig. 5(a) shows the results of our analysis for restoration schemes with different values of offset time for path (A_Path) as well as sub-path (A_Sub) restoration schemes. The results indicate, as expected, sub-path restoration scheme offers better performance, as compared to path restoration schemes. Fig. 5(b) presents

a comparison between analytical (A_Path, A_Sub) and simulation (S_Path, S_Sub) results for path restoration schemes. As evident from the plots, the simulation results are found in close agreement with the results of numerical analysis. We therefore extend the same simulation methodology is examine the performance of the proposed scheme.

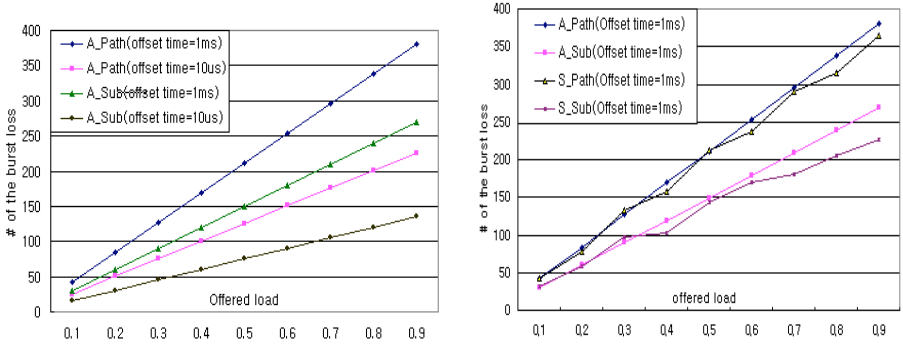


Fig. 5. The number of bursts lost taking into consideration the restoration scheme ((a) numerical analysis results, (b) simulation results)

Fig. 6 shows the plots of number of bursts lost vs. offered load for the proposed restoration scheme for threshold reaction times ranging from $10\,\mu s$ to $1\,ms$ at upstream node. In Fig. 6(a), the network uses an offset time=$1\,ms$. As evident from the plots, the number of bursts lost in the case of the conventional path restoration scheme is more than that occurs in the proposed restoration scheme. This is expected because in the latter case the upstream node can recreate the BCP and the burst can be sent to the destination node safely, if the threshold time is less than the time remaining before the arrival of the BDF at the upstream node. Therefore, by using the present restoration scheme, combined use of sub-path and path restoration with extended offset time, the upstream node can reroute incoming bursts until the source node receive the FIS message and, this lead to a much better performance.

As shown in Fig. 6(a), the number of bursts lost decreases as the threshold reaction time decreases. This is expected because the upstream node can reroute a larger number of bursts when the threshold reaction time is smaller. In Fig. 6(b), the offset time is $10\,\mu s$. Due to the smaller offset time, the proposed restoration scheme found to be less affected by the threshold time.

Fig.7 shows a comparison of the number of bursts lost between the proposed restoration scheme and the conventional restoration scheme for offset times of $1\,ms$ and $10\,\mu s$. The results clearly indicate show that the proposed restoration scheme is very effective in reducing burst losses.

In OBS networks, when link failure occurs, not only will the BCPs and BDFs present in the failed link be lost, but also those BDFs whose BCPs have already passed through the failed link. Due to the operational characteristics of the OBS, the num-

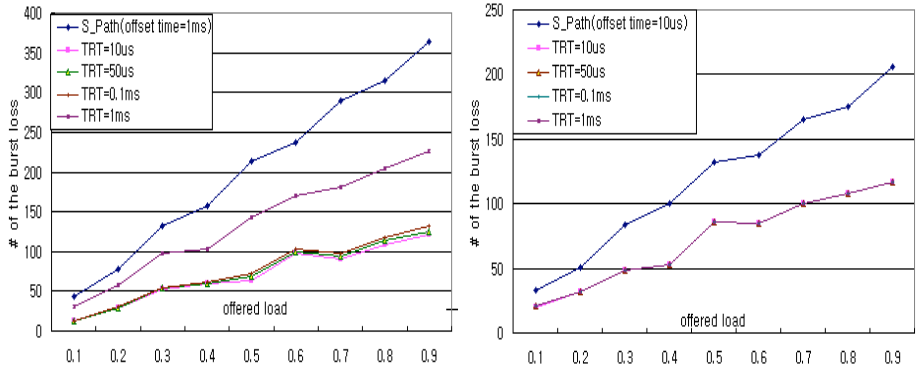


Fig. 6. The number of bursts lost vs. the offered load for various threshold times ((a) offset time: $1\,ms$, (b) offset time: $10\ \mu s$)

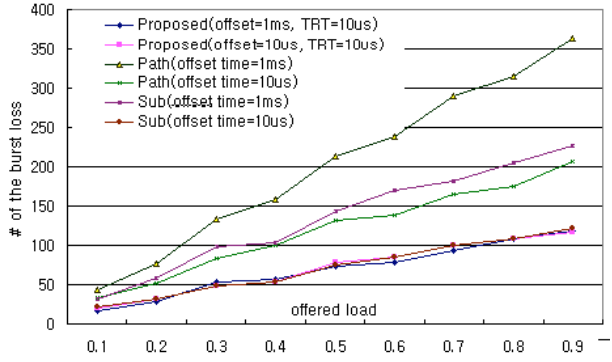


Fig. 7. Comparison of the number of bursts lost for proposed restoration schemes

ber of bursts lost is indeed dependent on the offset time for a given value of threshold reaction time of the networks. Therefore, in an OBS network, it is very important to select the appropriate offset time and restoration scheme in order to obtain the required QoS. Our future work will address this issue and explore further on QoS-aware restoration schemes for different service classes

6 Conclusion

In OBS networks, when a BCP arrives at an intermediate node, it is converted into electrical domain to process the BCP for the routing of the corresponding BDF and onward reservation. Following the processing of the BCP and subsequent switch reconfiguration during the stipulated offset time, the BDF can pass through the pre-configured optical switching unit without any O/E/O conversion. However, due to these operational characteristics of OBS network, when link failure occurs, factors

such as QoS can be seriously affected. Therefore, in this paper, we have examined the burst loss phenomenon occurring due to link failures in OBS network and proposed a new restoration scheme employing a combination of sub-path and path restoration schemes. The proposed scheme, in particular, employs sub-path restoration (as the first step) and path restoration (next step) scheme from upstream and source nodes respectively. However, sub-path restoration is employed by upstream nodes only when the remaining time left for yet-to-arrive BDFs are less than or equal to the threshold reaction time of the network. Our results indicate that the proposed scheme offers a significant improvement over the existing schemes, reported in the literature. In our future work, we would extend this scheme to design more robust QoS-aware restoration schemes for different service classes.

Acknowledgement. This work was supported by grant No. R05-2003-000-12183-0 from KOSEF and Joint Research Project under the KOSEF-NSF cooperative program and by the KOSEF through OIRC project.

References

1. C. Qiao & M. Yoo, Optical burst switching (OBS) – A new paradigm for an optical internet, Journal of High Speed Networks, 8(1), 1999, 69-84.
2. M. Yoo and C. Qiao, "QoS performance of optical burst switching in IP over WDM networks, " IEEE J. Selected Areas in Communications, vol.18, no. 10, pp. 2062-2071, Oct, 2000.
3. S. Junghans and C. M. Gauger, “Resource reservation in optical burst switching: architectures and realizations for reservation modules,” in Proc. OptiComm’2003, Dallas, TX, Oct, 2003.
4. S. Ramamurthy and B. Mukherjee, "Survivable WDM mesh networks, part 1-protection," Proc. IEEE INFOCOM'99, vol. 2, pp. 744-751, March 1999.
5. S. Ramamurthy and B. Mukherjee, "Survivable WDM mesh networks, part 2-protection," Proc. IEEE INFOCOM'99, vol. 2, pp. 2023-2030, March 1999.
6. J. Wang, L. Sahasrabuddhe & B. Mukherjee, “Path vs. Sub-Path vs. Link Restoration for Fault Management in IP-over-WDM Networks: Performance Comparisons Using GMPLS Control Signaling,” IEEE communication magazine, Nov. 2002.
7. L. Sahasrabuddhe , S. Ramamurthy and B. Mukherjee, “Fault management in IP-over-WDM networks: WDM protection versus IP restoration,” IEEE J. Selected Areas in Communications, vol.18, no. 1, pp. 21-23, Jan, 2002.
8. C. Assi, Y. Ye, A. Shami, S. Dixit, and M. Ali, “A hybrid distributed fault-management protocol for combating single-fiber failures mesh-based DWDM,” GLOBCOM, pp. 2676-2680, Nov, 2002.

Author Index

Abawajy, J.H. II-107
Abawajy, Jemal II-87
Abdullah, Azizol II-146
Abellanas, Manuel III-1, III-22
Acciani, Giuseppe II-979
Acosta-Elías, Jesús IV-177
Aggarwal, J.K. IV-311
Ahmad, Muhammad Bilal IV-877, IV-940, IV-948
Ahn, Byoungchul III-566, III-993
Ahn, Byungjun I-1125
Ahn, In-Mo IV-896
Ahn, Jaemin III-847
Ahn, JinHo III-376, IV-233
Ahn, Kiok I-1044
Ahn, ManKi I-517
Ahn, Seongjin I-142, I-1078
Ahn, Sung IV-489
Ahn, Yonghak I-1044
Ahn, Young Soo II-1079
Albertí, Margarita II-328, II-374
Albrecht, Andreas A. III-405
Alcaide, Almudena I-851
Alegre, David III-857
Aleixos, Nuria II-613
Alinchenko, M.G. III-217
Amaya, Jorge II-603
An, Beongku IV-974
An, Changho I-25
An, Ping IV-243
Anido, Luis II-922
Anikeenko, A.V. III-217
Annibali, Antonio III-722
Apu, Russel A. II-592
Asano, Tetsuo III-11
Atiqullah, Mir M. III-396
Attiya, Gamal II-97
Aung, Khin Mi Mi IV-574

Bachhiesl, Peter III-538
Bae, Hae-Young I-222, II-1079
Bae, Ihn-Han I-617
Bae, Sang-Hyun I-310, II-186, IV-359
Baik, Kwang-ho I-988
Baik, Ran III-425
Baik, Sung III-425, IV-206, IV-489
Bajuelos, António Leslie III-117, III-127
Bala, Jerzy IV-206, IV-489
Bang, Young-Cheol I-1125, II-913, IV-56
Bang, Young-Hwan I-491
Barel, Marc Van II-932
Barenco Abbas, Clàudia Jacy I-868
Barua, Sajib III-686
Becucci, M. II-374
Bekker, Henk III-32
Bellini, Francesco III-722
Beltran, J.V. II-631
Bencsura, Ákos II-290
Bertazzon, Stefania II-998
Bhatt, Mehul III-508
Bollman, Dorothy III-481, III-736
Boluda, Jose A. IV-887
Bonetto, Paola II-505
Bonitz, M. II-402
Borgosz, Jan III-715, IV-261
Borruso, Giuseppe II-1009, II-1089
Bose, Prosenjit III-22
Botana, F. II-761
Brass, Peter III-11
Brink, Axel III-32
Broeckhove, Jan IV-514
Brunelli, Roberto II-693
Bruno, D. II-383
Bruschi, Viola II-779
Bu, Jiajun III-886, IV-406
Bücker, H. Martin II-882
Buliung, Ronald N. II-1016
Buono, Nicoletta Del II-961, II-988
Buyya, Rajkumar IV-147
Byun, Kijong II-809

Cacciatore, M. II-366
Caeiro, Manuel II-922
Camp, Ellen Van II-932
Campa, S. II-206
Campos-Canton, Isaac IV-177
Capitelli, Francesco II-338
Capitelli, M. II-383

Carbonell, Mildrey I-903
Carretero, Jesús IV-496
Carvalho, Sílvia II-168
Casas, Giuseppe Las II-1036
Cendrero, Antonio II-779
Čermák, Martin III-325
Cha, Eui-Young II-486, IV-421
Cha, JeongHee I-17, I-41
Cha, Joo-Heon II-573
Chae, Jongwoo III-965, IV-983
Chae, Kijoon I-673
Chae, Oksam I-1044
Chambers, Desmond II-136
Chang, Beom H. I-191, I-693, IV-681
Chang, Byeong-Mo I-106
Chang, Hoon I-73
Chang, Min Hyuk IV-877
Chang, Yongseok IV-251
Chelli, R. II-374
Chen, Chun III-886, IV-406
Chen, Deren II-158
Chen, Tzu-Yi IV-20
Chen, Yen Hung III-355
Chen, Zhenming III-277
Cheng, Min III-729
Cheung, Chong-Soo I-310
Cheung, Wai-Leung II-246
Chi, Changkyun IV-647
Cho, Cheol-Hyung II-554, III-53
Cho, Chung-Ki III-847, III-926
Cho, Dong-Sub III-558
Cho, Haengrae III-548, III-696
Cho, Hanjin I-1007
Cho, Jae-Hyun II-486, IV-421
Cho, Jeong-Hyun IV-359
Cho, Jung-Hyun IV-251
Cho, Kyungsan I-167
Cho, Mi Gyung I-33
Cho, Seokhyang I-645
Cho, SungEon I-402
Cho, TaeHo I-567
Cho, We-Duke I-207, I-394
Cho, Yongsun I-426
Cho, Yookun I-547, I-978, IV-799
Cho, Youngjoo IV-647
Cho, Youngsong II-554, III-62
Choi, Chang-Gyu IV-251
Choi, Chang-Won I-302
Choi, Changyeol I-207
Choi, Dong-Hwan III-288
Choi, Doo Ho I-1151
Choi, Eun-Jung I-683
Choi, Eunhee II-913
Choi, Hoo-Kyun IV-11
Choi, Hoon II-196
Choi, HyungIl I-17, I-41
Choi, Joonsoo III-837
Choi, Kee-Hyun I-434
Choi, SangHo IV-29
Choi, Sung Jin IV-637
Choi, Tae-Sun IV-271, IV-291, IV-338, IV-348, IV-877
Choi, Uk-Chul IV-271
Choi, Won-Hyuck IV-321, IV-451
Choi, Yong-Soo I-386
Choi, Yoon-Hee IV-271, IV-338, IV-348
Choi, YoungSik I-49, II-942
Choi, Yumi I-663
Choirat, Christine III-298
Chong, Kiwon I-426
Choo, Hyunseung I-360, I-663, I-765, III-315, IV-56, IV-431
Choo, Kyonam III-585
Chover, M. II-622, II-703
Choy, Yoon-Chul IV-743, IV-772
Chu, Jie II-126
Chun, Jong Hun IV-940
Chun, Junchul I-25
Chun, Myung Geun I-635, IV-828, IV-924
Chung, Chin Hyun I-1, I-655, IV-964
Chung, Ilyong II-178, IV-647
Chung, Jin Wook I-142, I-1078
Chung, Min Young I-1159, IV-46
Chung, Mokdong I-537, III-965, IV-983
Chung, Tai-Myung I-183, I-191, I-238, I-693, IV-681
Cintra, Marcelo III-188
Clifford, Gari I-352
Collura, F. II-536
Contero, Manuel II-613
Costa Sousa, Mario III-247
Crane, Martin III-473
Crocchianti, Stefano II-422
Crothers, D.S.F. II-321
Cruz R., Laura III-415, IV-77

Cruz-Chavez, Marco Antonio IV-553
Cutini, V. II-1107
Cyganek, Bogusław III-715, IV-261

D'Amore, L. II-515
Dağ, Hasan III-795
Daly, Olena IV-543
Daněk, J. II-456
Danelutto, M. II-206
Das, Sandip III-42
Datta, Amitava IV-479
Datta, Debasish IV-994
Delaitre, T. II-30
Demidenko, Eugene IV-933
Denk, F. II-456
Díaz, José Andrés III-158
Díaz-Báñez, Jose Miguel III-99, III-207
Díaz-Verdejo, Jesús E. I-841
Diele, Fasma II-932, II-971
Discepoli, Monia III-745, IV-379
Djemame, Karim II-66
Dong, Zhi II-126
Dózsa, Gábor II-10
Duato, J. II-661
Durán, Alfonso I-949, III-857

Effantin, Brice III-648
Eick, Christoph F. IV-185
Engel, Shane II-1069
Eom, Sung-Kyun IV-754
Ercan, M. Fikret II-246
Erciyes, Kayhan III-518, III-528
Esposito, Fabrizio II-300
Estévez-Tapiador, Juan M. I-841
Estrada, Hugo IV-506, IV-783
Eun, Hye-Jue I-122

Fan, Kaiqin II-126
Farias, Cléver R.G. de II-168
Faudot, Dominique III-267
Feng, Yu III-498
Fernández, Marcos II-661, II-671
Fernández-Medina, Eduardo I-968
Ferrer-Gomila, Josep Lluís I-831, I-924, IV-223
Filinov, V. II-402
Fiori, Simone II-961
Flahive, Andrew III-508
Formiconi, Andreas Robert II-495
Fornarelli, Girolamo II-979
Fortov, V. II-402
Foster, Kevin III-247
Fragoso Diaz, Olivia G. IV-534, IV-808
Fraire H., Héctor III-415, IV-77
Frausto-Solís, Juan III-415, III-755, IV-77, IV-553
Fung, Yu-Fai II-246

Galpert, Deborah I-903
Gálvez, Akemi II-641, II-651, II-771, II-779
Gameiro Henriques, Pedro II-817
García, Alfredo III-22
Garcia, Ernesto II-328
García, Félix IV-496
García, Inmaculada III-877
García, José Daniel IV-496
García-Teodoro, Pedro I-841
Gardner, Henry III-776
Gavrilova, Marina L. II-592, III-217
Gerace, Ivan III-745, IV-379
Gerardo, Bobby D. I-97
Gervasi, Osvaldo II-827, II-854
Giansanti, Roberto III-575
Go, Hyoun-Joo IV-924
Gola, Mariusz III-611
Gómez, Francisco III-207
González Serna, Juan G. IV-137
Gourlay, Iain II-66
Goyeneche, A. II-30
Gregori, Stefano II-437
Grein, Martin II-843
Guan, Jian III-706
Guarracino, Mario R. II-505, II-515
Gulbag, Ali IV-389
Guo, Wanwu IV-471, IV-956
Guo, Xinyu II-751
Gupta, Sudhir IV-791
Gutiérrez, Carlos I-968
Gutiérrez, Miguel III-857

Ha, Eun-Ju IV-818
Ha, JaeCheol I-150
Ha, Jong-Eun IV-896, IV-906, IV-915
Ha, Kyeoung Ju IV-196
Ha, Yan I-337
Hackman, Mikael I-821
Hahn, Kwang-Soo III-837

Hamam, Yskandar II-97
Hamdani, Ajmal H. II-350
Han, Dongsoo IV-97
Han, Jongsu III-955
Han, Qianqian II-272
Han, Seok-Woo I-122
Han, Seung Jo IV-948
Han, Sunyoung I-1115
Han, Tack-Don II-741
Han, Young J. I-191, I-693, IV-681
Haron, Fazilah IV-147
Healey, Jennifer I-352
Heo, Joon I-755
Herges, Thomas III-454
Hernández, Julio César I-812, I-851, I-960
Hiyoshi, Hisamoto III-71
HlaváII-ček, I. II-456
Hlavaty, Tomas III-81
Hoffmann, Kenneth R. III-277
Hong, Choong Seon I-755, I-792, I-915, I-1134
Hong, Chun Pyo III-656, IV-106
Hong, Dong Kwon I-134
Hong, Hyun-Ki II-799
Hong, Inki I-1125
Hong, Kwang-Seok I-89, IV-754
Hong, Man-Pyo IV-611
Hong, Manpyo III-867, IV-708
Hong, Maria I-57
Hong, Seong-sik I-1060
Hong, Suk-Ki II-902, II-913
Hong, Youn-Sik III-1002
Hosseini, Mohammad Mahdi III-676
Hruschka, Eduardo R. II-168
Hu, Hualiang II-158
Hu, Weixi II-751
Huang, Changqin II-158
Huettmann, Falk II-1117
Huguet-Rotger, Llorenç I-831, IV-223
Huh, Eui-Nam I-370, I-738, I-746
Hur, Hye-Sun III-1002
Hurtado, Ferran III-22
Hwang, Byong-Won III-386, IV-281
Hwang, Chan-Sik III-288
Hwang, Chong-Sun I-286, III-945, IV-233, IV-584
Hwang, EenJun IV-838, IV-859
Hwang, Ha Jin I-577
Hwang, Jun I-1, I-655, I-746
Hwang, Seong Oun II-46
Hwang, Sun-Myung I-481
Hwang, Sungsoon II-1026
Hwang, Yong Ho I-442
Hwang, Yong-Ho II-799
Hwang, YoungHa IV-460

Ibrahim, Hamidah II-146
Iglesias, A. II-641, II-651, II-771
Im, Chaetae I-246
Im, Jae-Yuel IV-655
In, Chi Hyung I-792
Inguglia, Fabrizio II-505
Izquierdo, Antonio I-812

Jabbari, Arash II-432
Jacobs, Gwen III-257
Jang, HyoJong I-41
Jang, Jong-Soo I-988, IV-594
Jang, Jongsu I-776
Jang, Kyung-Soo I-434
Jang, Min-Soo III-489
Jang, Sang-Dong II-216
Jang, Seok-Woo I-9
Jang, Tae-Won I-386
Je, Sung-Kwan IV-421, II-486
Jedlovszky, P. III-217
Jeon, Hoseong I-765
Jeon, Jaeeun III-566
Jeong, Chang Yun I-337
Jeong, Chang-Sung I-319, II-789
Jeong, Eunjoo I-418
Jeong, Hae-Duck J. III-827
Jeong, Ok-Ran III-558
Jeong, Sam Jin IV-213
Jiang, Minghui III-90
Jin, Guiyue III-993
Jin, Hai II-116, II-126
Jin, Min IV-763, IV-849
Jin, Zhou II-272
Jo, Hea Suk I-711, III-1010
Jo, Jang-Wu I-106
Jo, Sun-Moon IV-524
Jonsson, Erland I-821
Jonsson, Håkan III-168
Joo, Pan-Yuh I-394
Jorge, Joaquim II-613

Jun, Woochun II-902, II-913
Jung, Changryul I-294
Jung, Il-Hong I-451
Jung, Kyung-Yong II-863
Jung, Yoon-Jung I-491

Kacsuk, Péter II-10, II-37, II-226
Kanaroglou, Pavlos S. II-1016
Kang, Chang Wook II-554
Kang, Dong-Joong IV-896, IV-906, IV-915
Kang, Euisun I-57
Kang, HeeGok I-402
Kang, Ho-Kyung III-602
Kang, Ho-Seok I-1105
Kang, Hyunchul I-345
Kang, Kyung-Pyo IV-348
Kang, KyungWoo I-65
Kang, Min-Goo I-302, I-386, I-394
Kang, SeokHoon I-270, III-585
Kang, Seung-Shik IV-735
Kang, Sunbu III-926
Kang, Sung Kwan IV-940
Kang, Sungkwon III-847, IV-11
Kang, Tae-Ha IV-281
Kang, Won-Seok IV-167
Kasahara, Yoshiaki I-915
Kasprzak, Andrzej III-611
Kaußner, Armin II-843
Kelz, Markus III-538
Kheddouci, Hamamache III-267
Kim, Backhyun I-345
Kim, Bonghan I-1007
Kim, Byoung-Koo I-998, IV-594
Kim, Byunggi I-418
Kim, Byungkyu III-489
Kim, Chang Hoon III-656, IV-106
Kim, Chang-Soo I-410
Kim, ChangKyun I-150
Kim, Changnam I-738
Kim, ChaYoung IV-233
Kim, Cholmin III-867
Kim, D.S. I-183
Kim, Dae Sun I-1134
Kim, Dae-Chul IV-271
Kim, Daeho I-1078
Kim, Deok-Soo II-554, II-564, II-583, III-53, III-62
Kim, Dohyeon IV-974
Kim, Dong S. I-693, IV-681
Kim, Dong-Hoi I-81
Kim, Dong-Kyoo III-896, III-906, IV-611
Kim, Dongho I-57
Kim, Donguk III-62
Kim, Duckki I-378
Kim, Gwang-Hyun I-1035
Kim, Gyeyoung I-9, I-17, I-41
Kim, Haeng-Kon I-461
Kim, Haeng-kon IV-717
Kim, Hak-Ju I-238
Kim, Hak-Keun IV-772
Kim, Hangkon I-587
Kim, Hanil II-892
Kim, Hie-Cheol II-20
Kim, Hiecheol III-656
Kim, Ho J. IV-791
Kim, Hyeong-Ju I-998
Kim, Hyun Gon I-1151
Kim, Hyun-Sung IV-617
Kim, Hyuncheol I-1078
Kim, Hyung-Jong I-567, I-683
Kim, Ik-Kyun I-998, IV-594
Kim, Iksoo I-270, I-345
Kim, Injung I-491
Kim, Jae-Kyung IV-743
Kim, Jaehyoun I-360
Kim, Jay-Jung II-573
Kim, Jeeyeon I-895
Kim, Jeom Goo I-1026
Kim, Jin I-81
Kim, Jin Geol IV-29
Kim, Jin Ok I-1, I-655, IV-964
Kim, Jin Soo IV-964
Kim, Jong G. II-1
Kim, Jong-bu IV-725
Kim, Jong-Woo I-410
Kim, Joo-Young IV-338
Kim, JoonMo I-567
Kim, Jung-Sun I-175, III-985, IV-321, IV-451
Kim, Jung-Woo II-741
Kim, Kee-Won IV-603, IV-672
Kim, Keecheon I-1115
Kim, Ki-Hyung IV-167
Kim, Ki-Tae IV-524
Kim, Ki-Young I-988, IV-594

Kim, KiIl IV-460
Kim, KiJoo I-49
Kim, Kweon Yang I-134
Kim, Kyungsoo II-467
Kim, Mansoo I-537
Kim, Mi-Ae I-159, I-722
Kim, Mi-Jeong I-394
Kim, Mihui I-673
Kim, Min-Su I-1159
Kim, Minsoo I-175, I-230
Kim, Misun I-199, I-262
Kim, Miyoung I-199, I-262
Kim, MoonJoon I-73
Kim, Moonseong IV-56
Kim, Myuhng-Joo I-683
Kim, Nam-Chang I-1105
Kim, Nam-Yeun IV-87
Kim, Pan Koo IV-940
Kim, Pankoo II-892
Kim, Pyung Soo III-975, IV-301
Kim, Sang Ho I-608, I-1069
Kim, SangHa IV-460
Kim, Sangkyun I-597
Kim, Seokyu I-150
Kim, Seong-Cheol III-837
Kim, Seonho I-328
Kim, Seungjoo I-645, I-895
Kim, Shin-Dug II-20
Kim, Soon Seok I-215
Kim, Soon-Dong IV-611
Kim, Soung Won I-577
Kim, Su-Hyun I-1035
Kim, Sung Jo I-278
Kim, Sung Ki I-246
Kim, Sung Kwon I-215
Kim, Sung-Ho IV-251
Kim, Sung-Hyun I-150
Kim, Sung-Min III-602
Kim, Sung-Ryul III-367
Kim, Sung-Suk IV-924
Kim, Sunghae I-1078
Kim, Sungsoo I-207
Kim, SungSuk I-286
Kim, Tae-Kyung I-238
Kim, Taekkeun III-926
Kim, Tai-Hoon I-451, I-461, I-1052, IV-717
Kim, Won I-17
Kim, Wonil III-896, III-906
Kim, Woo-Hun IV-617
Kim, Wu Woan II-216, II-262
Kim, Yong-Guk III-489
Kim, Yong-Sung I-122, I-337
Kim, Yoon Hyuk II-467
Kim, Young Kuen III-975
Kim, Young-Chon IV-994
Kim, Young-Sin I-738, I-746
Kim, YounSoo II-196
Kiss, T. II-30
Kizilova, Natalya II-476
Ko, Myeong-Cheol IV-772
Ko, Younghun I-360
Kóczy, László T. I-122
Koh, JinGwang I-294, I-310, I-402
Koh, Kwang-Won II-20
Kolingerová, Ivana II-544, II-682, III-198
Koo, Han-Suh II-789
Kouadri Mostéfaoui, Ghita I-537, III-965, IV-983
Kouh, Hoon-Joon IV-524
Ku, Kyo Min IV-196
Kulikov, Gennady Yu. III-345, III-667
Kwak, JaeMin I-402
Kwak, Jin I-895, III-955
Kwak, Keun Chang I-635, IV-828, IV-924
Kwon, Chang-Hee I-310
Kwon, Ki Jin I-1159
Kwon, Kyohyeok I-142
Kwon, Soonhak III-656, IV-106
Kwon, Taekyoung I-728
Kwon, Yong-Won I-319
Kwon, YongHoon III-847, III-926

Laccetti, G. II-515, II-525
Laganà, Antonio II-328, II-357, II-374, II-422, II-437, II-827, II-854
Lagzi, István II-226
Lang, Bruno II-882
Lara, Sheila L. Delfín IV-808
Lau, Matthew C.F. II-873
Lázaro, Miguel II-779
Lee, Bo-Hyeong IV-46
Lee, Bong Hwan I-352
Lee, Bum Ro IV-964
Lee, Byong Gul I-134

Lee, Byong-Lyol I-663
Lee, Byung Kwan I-33
Lee, Byung-Wook I-746
Lee, Byunghoon III-53
Lee, Dae Jong I-635, IV-828
Lee, Dea Hwan I-915
Lee, Deok-Gyu IV-66
Lee, Dong Chun I-1052, I-1097
Lee, Dongkeun I-1115
Lee, Dongryeol I-510
Lee, Eun-ser I-451
Lee, Gang-Soo I-491
Lee, Gunhee III-906
Lee, Gunhoon III-566
Lee, Hae-Joung IV-994
Lee, Hae-ki IV-725
Lee, Han-Ki I-159
Lee, Ho-Dong III-489
Lee, HongSub I-567
Lee, HoonJae I-517
Lee, Hunjoo II-809, II-837
Lee, Hwang-Jik II-20
Lee, Hyon-Gu I-89
Lee, Hyun Chang II-186
Lee, HyunChan II-554, II-564
Lee, Hyung-Woo I-302, I-386
Lee, HyungHyo I-701
Lee, Im-Yeong I-557, III-1020, IV-66
Lee, In Hwa I-278
Lee, In-Ho II-573
Lee, Jae Kwang I-254, I-1007
Lee, Jae-il I-728
Lee, Jaeheung I-547
Lee, Jaeho II-564
Lee, Jaewan I-97
Lee, Jong Sik III-621, III-630
Lee, Jong-Suk Ruth III-827
Lee, Joongjae I-17
Lee, Ju-Hyun IV-11
Lee, Jung-Hyun II-863
Lee, Jungsik I-97
Lee, KangShin I-567
Lee, Keon-Jik III-638
Lee, Key Seo IV-964
Lee, Ki Dong III-566, III-993
Lee, Kwan H. III-178
Lee, Kwang-Ok I-310
Lee, Kwnag-Jae IV-451
Lee, Kyong-Ho IV-743
Lee, Kyung Whan I-451
Lee, Malrey I-97
Lee, Myung Eui III-975, IV-301
Lee, Myung-Sub IV-441
Lee, Namhoon I-491
Lee, Okbin II-178
Lee, Ou-Seb I-394
Lee, Pil Joong I-442, I-471, I-802
Lee, Sang Hyo IV-964
Lee, Sang-Hak III-288
Lee, Sang-Ho IV-689
Lee, Sangkeon I-1017, I-1088
Lee, SangKeun I-286
Lee, Seok-Joo III-489
Lee, Seung IV-725
Lee, SeungYong I-701
Lee, Soo-Gi I-625
Lee, SooCheol IV-838, IV-859
Lee, Soung-uck III-867
Lee, Sung-Woon IV-617
Lee, Sungchang I-1125
Lee, Sungkeun I-294
Lee, Tae-Jin I-1159, IV-46
Lee, Tae-Seung III-386, IV-281
Lee, Taehoon II-178
Lee, Tong-Yee II-713, II-721
Lee, Won Goo I-254
Lee, Won-Ho III-638
Lee, Won-Hyung I-159, I-722
Lee, Won-Jong II-741
Lee, Woojin I-426
Lee, Woongjae I-1, I-655
Lee, YangKyoo IV-838, IV-859
Lee, Yeijin II-178
Lee, YoungSeok II-196
Lee, Yugyung I-410
Leem, Choon Seong I-597, I-608, I-1069
Lendvay, György II-290
Levashov, P. II-402
Lho, Tae-Jung IV-906
Li, Chunlin IV-117
Li, Gang II-252
Li, Layuan IV-117
Li, Mingchu II-693
Li, Shengli II-116
Li, Xiaotu II-252
Li, Xueyao IV-414

Li, Yufu II-116
Lim, Heeran IV-708
Lim, Hwa-Seop I-386
Lim, Hyung-Jin I-238
Lim, Joon S. IV-791
Lim, SeonGan I-517
Lim, Soon-Bum IV-772
Lim, Younghwan I-57
Lin, Hai II-236
Lin, Ping-Hsien II-713
Lin, Wenhao III-257
Lindskog, Stefan I-821
Lísal, Martin II-392
Liturri, Luciano II-979
Liu, Da-xin III-706
Liu, Yongle III-498
Llanos, Diego R. III-188
Lombardo, S. II-1046
Longo, S. II-383
Lopez, Javier I-903
López, Mario Alberto III-99
Lovas, Róbert II-10, II-226
Lu, Chaohui IV-243
Lu, Jianfeng III-308
Lu, Yilong III-729
Lu, Yinghua IV-956
Lu, Zhengding IV-117
Luna-Rivera, Jose Martin IV-177
Luo, Yingwei III-335

Ma, Zhiqiang IV-471
Machì, A. II-536
Maddalena, L. II-525
Maponi, Pierluigi III-575
Marangi, Carmela II-971
Mariani, Riccardo III-745
Marinelli, Maria III-575
Mark, Christian II-843
Marques, Fábio III-127
Marshall, Geoffrey III-528
Martínez, Alicia IV-506, IV-783
Martoyan, Gagik A. II-313
Mastronardi, Nicola II-932
Matsuhisa, Takashi III-915
Maur, Pavel III-198
Medvedev, N.N. III-217
Mejri, Mohamed I-938
Melnik, Roderick V.N. III-817
Ménegaux, David III-267
Merkulov, Arkadi I. III-667
Merlitz, Holger III-465
Messelodi, Stefano II-693
Miguez, Xochitl Landa IV-137
Milani, Alfredo III-433, IV-563
Min, Byoung Joon I-246
Min, Hongki III-585
Min, Jun Oh I-635, IV-828
Min, Young Soo IV-869
Minelli, P. II-383
Ming, Zeng IV-127
Mitrani, I. II-76
Moh, Sangman IV-97
Molina, Ana I. III-786
Mollá, Ramón III-877
Monterde, J. II-631
Moon, Aekyung III-696
Moon, Kiyoung I-776
Moon, SangJae I-150, I-517
Moon, Young-Jun I-1088
Mora, Graciela IV-77
Moradi, Shahram II-432
Moreno, Oscar III-481
Moreno-Jiménez, Carlos III-1
Morici, Chiara III-433
Morillo, P. II-661
Mukherjee, Biswanath IV-994
Mumey, Brendan III-90
Mun, Youngsong I-199, I-262, I-378, I-738, I-1144
Murgante, Beniamino II-1036
Murli, A. II-515
Murri, Roberto III-575
Muzaffar, Tanzeem IV-291

Na, Jung C. I-191, I-693, IV-681
Na, Won Shik I-1026
Na, Young-Joo II-863
Nam, Dong Su I-352
Nam, Junghyun I-645
Nandy, Subhas C. III-42
Navarro-Moldes, Leandro IV-177
Naya, Ferran II-613
Nedoma, Jiří II-445, II-456
Neelamkavil, Francis II-741, IV-743
Németh, Csaba II-10
Nguyen, Thai T. IV-791
Nicotra, F. II-536

Nielsen, Frank III-147
Niewiadomski, Radoslaw III-433
Nishida, Tetsushi III-227
Nock, Richard III-147
Noh, Bong-Nam I-175, I-230
Noh, BongNam I-701
Noh, JiSung II-942
Noh, SungKee IV-460
Noltemeier, Hartmut II-843

O'Loughlin, Finbarr II-136
O'Rourke, S.F.C. II-321
Oh, Am Sok I-33
Oh, ByeongKyun I-527, IV-698
Oh, Jai-Ho I-765
Oh, Kyu-Tae III-985
Oh, Soohyun III-955
Oh, Sun-Jin I-617
Oh, Wongeun I-294
Oh, Young-Hwan I-222
Ohn, Kyungoh III-548
Olanda, Ricardo II-671
Oliveira Albuquerque, Robson de I-868
Onieva, Jose A. I-903
Orduña, J.M. II-661
Orozco, Edusmildo III-481, III-736
Orser, Gary III-257
Ortega, Manuel III-786
Otero, César II-641, II-779, III-158
Othman, Abdulla II-66
Othman, Abu Talib II-146
Othman, Mohamed II-146
Ouyang, Jinsong I-345
Ozturk, Zafer Ziya IV-398

Pacifici, Leonardo II-357
Pakdel, Hamid-Reza III-237
Palladini, Sergio II-1057
Palmer, J. II-76
Palmieri, Francesco I-882
Palop, Belén III-188
Pan, Zhigeng II-236, II-731, II-751, III-308
Pardo, Fernando IV-887
Park, Chang Won IV-627
Park, Chang-Hyeon IV-441
Park, Dong-Hyun II-863
Park, Goorack I-25
Park, Gwi-Tae III-489
Park, Gyung-Leen I-114
Park, Hee-Un I-557
Park, Hong Jin I-215
Park, Hyoung-Woo I-319, II-1, III-827
Park, Hyunpung III-178
Park, IkSu I-527, IV-698
Park, JaeHeung I-73
Park, Jaehyung I-1159
Park, Jihun IV-311, IV-369
Park, Jong An IV-877, IV-940, IV-948
Park, Jong Sou IV-574
Park, Jongjin I-1144
Park, Joo-Chul I-9
Park, Joon Young II-554, II-564
Park, Jun-Hyung I-230
Park, Ki heon IV-29
Park, Kyeongmo I-500
Park, Kyung-Lang II-20
Park, M.-W. II-573
Park, Mingi I-97
Park, Namje I-776
Park, Sangjoon I-418
Park, Seong-Seok I-410
Park, Seung Jin IV-877, IV-948
Park, SeungBae I-527, IV-698
Park, Sihn-hye III-896
Park, Soohong III-975
Park, Soon-Young II-1079
Park, Sunghun IV-311, IV-369
Park, Taehyung I-1017, I-1088
Park, Taejoon II-837
Park, Woo-Chan II-741
Park, Yongsu I-547, I-978, IV-799
Pastor, Oscar IV-506, IV-783
Payeras-Capella, Magdalena I-831, IV-223
Pazos R., Rodolfo A. III-415, IV-77
Pedlow, R.T. II-321
Peña, José M. II-87
Pérez O., Joaquín III-415, IV-77
Pérez, José María IV-496
Pérez, María S. II-87
Pérez, Mariano II-671
Petri, M. II-1046, II-1107
Petrosino, A. II-525
Pfarrhofer, Roman III-538
Pflug, Hans-Joachim II-882
Piantanelli, Anna III-575

Piattini, Mario I-968
Pieretti, A. II-366
Piermarini, Valentina II-422
Pierro, Cinzia II-338
Pietraperzia, G. II-374
Pineda, Ulises IV-177
Ping, Tan Tien IV-147
Pişkin, Şenol III-795
Podesta, Karl III-473
Poggioni, Valentina IV-563
Politi, Tiziano II-961, II-988
Ponce, Eva I-949
Porschen, Stefan III-137
Puchala, Edward IV-39
Pugliese, Andrea II-55
Puig-Pey, J. II-651, II-771
Puigserver, Macià Mut I-924
Puttini, Ricardo S. I-868

Qi, Zhaohui II-252
Qin, Zhongping III-90

Ra, In-Ho I-310, IV-359
Radulovic, Nenad III-817
Ragni, Stefania II-971
Rahayu, Wenny III-443, III-508
Ramos, J.F. II-622, II-703
Ramos, Pedro III-22
Rebollo, C. II-703
Recio, T. II-761
Redondo, Miguel A. III-786
Reitsma, Femke II-1069
Remigi, Andrea III-745
Remolar, I. II-703
Rho, SeungMin IV-859
Ribagorda, Arturo I-812
Riganelli, Antonio II-374, II-827
Rivera-Campo, Eduardo III-22
Ro, Yong Man III-602
Robinson, Andrew III-443
Robles, Víctor II-87
Rodionov, Alexey S. III-315, IV-431
Rodionova, Olga K. III-315, IV-431
Rodríguez O., Guillermo III-415, IV-77
Rodríguez, Judith II-922
Rogerson, Peter II-1096
Roh, Sun-Sik I-1035
Roh, Yong-Wan I-89
Rosi, Marzio II-412
Rotger, Llorenç Huguet i I-924
Roy, Sasanka III-42
Rui, Zhao IV-127
Ruskin, Heather J. III-473, III-498
Rutigliano, M. II-366
Ryoo, Intae I-1026
Ryou, Hwang-bin I-1060
Ryou, Jaecheol I-776
Ryu, Eun-Kyung IV-603, IV-655, IV-665, IV-672
Ryu, So-Hyun I-319
Ryu, Tae W. IV-185, IV-791

Safouhi, Hassan II-280
Samavati, Faramarz F. III-237, III-247
Sampaio, Alcínia Zita II-817
Sánchez, Alberto II-87
Sánchez, Carlos II-328
Sánchez, Ricardo II-603
Sánchez, Teresa I-949
Sanna, N. II-366
Santaolaya Salgado, René IV-534, IV-808
Santos, Juan II-922
Santucci, A. II-1107
Sanvicente-Sánchez, Héctor III-755
Sasahara, Shinji III-11
Sastrón, Francisco III-857
Schoier, Gabriella II-1009, II-1089
Schug, Alexander III-454
Sellarès, Joan Antoni III-99
Senger, Hermes II-168
Seo, Dae-Hee I-557, III-1020
Seo, Heekyung III-837
Seo, Kyong Sok I-655
Seo, Seung-Hyun IV-689
Seo, Sung Jin I-1
Seo, Young Ro IV-964
Seong, Yeong Kyeong IV-338
Seri, Raffaello III-298
Seung-Hak, Rhee IV-948
Seznec, Andre I-960
Sgamellotti, Antonio II-412
Shahdin, S. II-350
Shen, Liran IV-414
Shen, Weidong IV-1
Shim, Hye-jin IV-321
Shim, Jae-sun IV-725
Shim, Jeong Min IV-869

Shim, Young-Chul I-1105
Shin, Byung-Joo IV-763, IV-849
Shin, Dong-Ryeol I-434
Shin, Hayong II-583
Shin, Ho-Jun I-625
Shin, Jeong-Hoon IV-754
Shin, Seung-won I-988
Shin, Yongtae I-328
Shindin, Sergey K. III-345
Sierra, José María I-851, I-812, I-960
Silva, Fabrício A.B. da II-168
Silva, Tamer Américo da I-868
Sim, Sang Gyoo I-442
Singh, Gujit II-246
Sipos, Gergely II-37
Skala, Václav III-81, III-325
Skouteris, Dimitris II-357
Slim, Chokri III-935
Smith, William R. II-392
So, Won-Ho IV-994
Sodhy, Gian Chand IV-147
Sohn, Sungwon I-776
Sohn, Won-Sung IV-743, IV-772
Sohn, Young-Ho IV-441
Song, Geun-Sil I-159, I-722
Song, Hyoung-Kyu I-386, I-394
Song, Il Gyu I-792
Song, Jin-Young II-799
Song, Kyu-Yeop IV-994
Song, Mingli III-886, IV-406
Song, Myunghyun I-294
Song, Seok Il IV-869
Song, Sung Keun IV-627
Song, Teuk-Seob IV-743
Sosa, Víctor J. Sosa IV-137
Soto, Leonardo II-603
Sousa Jr., Rafael T. de I-868
Soykan, Gürkan III-795
Stefano, Marco Di II-412
Stehlík, J. II-456
Stevens-Navarro, Enrique IV-177
Stögner, Herbert III-538
Strandbergh, Johan I-821
Studer, Pedro II-817
Sturm, Patrick III-109
Sug, Hyontai IV-158
Sugihara, Kokichi III-53, III-71, III-227
Sulaiman, Md Nasir II-146
Sun, Jizhou II-252, II-272

Tae, Kang Soo I-114
Talia, Domenico II-55
Tan, Rebecca B.N. II-873
Tang, Chuan Yi III-355
Taniar, David III-508, IV-543
Tasaltin, Cihat IV-398
Tasso, Sergio II-437
Tavadyan, Levon A. II-313
Techapichetvanich, Kesaraporn IV-479
Tejel, Javier III-22
Temurtas, Fevzullah IV-389, IV-398
Temurtas, Hasan IV-398
Thanh, Nguyen N. III-602
Thulasiram, Ruppa K. III-686
Thulasiraman, Parimala III-686
Togores, Reinaldo II-641, II-779, III-158
Tomás, Ana Paula III-117, III-127
Tomascak, Andrew III-90
Torres, Joaquín I-851
Torres-Jimenez, Jose IV-506
Trendafilov, Nickolay T. II-952
Turányi, Tamás II-226

Uhl, Andreas III-538
Uhmn, Saangyong I-81
Um, Sungmin I-57

Valdés Marrero, Manuel A. IV-137, IV-534, IV-808
Vanmechelen, Kurt IV-514
Vanzi, Eleonora II-495
Varnuška, Michal II-682
Vásquez Mendez, Isaac M. IV-534, IV-808
Vavřík, P. II-456
Vehreschild, Andre II-882
Ventura, Immaculada III-207
Verduzco Medina, Francisco IV-137
Ves, Esther De IV-887
Villalba, Luis Javier García I-859, I-868
Voloshin, V.P. III-217

Wang, Huiqiang IV-414
Wang, Tong III-706
Wang, Xiaolin III-335
Watson, Anthony IV-471
Wenzel, Wolfgang III-454, III-465
Willatzen, Morten III-817

Winter, S.C. II-30
Won, Dongho I-645, I-895, III-955
Woo, Yoseop I-270, I-345, III-585
Wouters, Carlo III-508
Wozniak, Michal III-593
Wu, Bang Ye III-355
Wu, Guohua II-731
Wyvill, Brian III-247

Xinyu, Yang IV-127
Xu, Guang III-277
Xu, Jinhui III-277
Xu, Qing II-693
Xu, Zhuoqun III-335

Yamada, Ikuho II-1096
Yan, Shaur-Uei II-721
Yang, Bailin II-236
Yang, Jin S. I-191, I-693, IV-681
Yang, Jong-Un IV-359
Yang, Shulin IV-1
Yang, Sun Ok I-286
Yang, Sung-Bong II-741, IV-743
Yang, SunWoong I-73
Yang, Tz-Hsien II-713
Yang, Zhiling II-126
Yao, Zhenhua III-729
Yap, Chee III-62
Yaşar, Osman III-795, III-807
Yavari, Issa II-432
Yen, Sung-Ming I-150
Yi, Myung-Kyu III-945, IV-584
Yi, Shi IV-127
Yim, Wha Young IV-964
Yin, Xuesong II-731
Yoe, Hyun I-294, I-402
Yong, Chan Huah IV-147
Yoo, Hyeong Seon I-510
Yoo, Jae Soo IV-869
Yoo, Kee-Young III-638, IV-87, IV-196, IV-603, IV-617, IV-655, IV-665, IV-672
Yoo, Kil-Sang I-159
Yoo, Kook-yeol IV-329
Yoo, Sang Bong II-1079
Yoo, Weon-Hee IV-524
Yoo, Wi Hyun IV-196
Yoon, Eun-Jun IV-665
Yoon, Hyung-Wook IV-46
Yoon, Jin-Sung I-9
Yoon, Ki Song II-46
Yoon, Miyoun I-328
You, Il-Sun I-167
You, Mingyu III-886, IV-406
You, Young-Hwan I-386, I-394
Youn, Chan-Hyun I-352
Youn, Hee Yong I-114, I-711, III-1010, IV-627, IV-637
Yu, Chansu IV-97
Yu, Kwangseok III-62
Yu, Qizhi II-236
Yum, Dae Hyun I-471, I-802
Yumusak, Nejat IV-389, IV-398
Yun, Byeong-Soo IV-818
Yun, Miso II-892

Zaia, Annamaria III-575
Zeng, Qinghuai II-158
Zhang, Hu III-764
Zhang, Jiawan II-252, II-272, II-693
Zhang, Jing IV-994
Zhang, Mingmin II-236
Zhang, Minming III-308
Zhang, Qin II-116
Zhang, Rubo IV-414
Zhang, Yi II-272
Zhang, Zhaoyang IV-243
Zhao, Chunjiang II-751
Zhou, Jianying I-903
Zhu, Binhai III-90, III-257
Zotta, D. II-1046