

Commenced Publication in 1973

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Editorial Board

David Hutchison

Lancaster University, UK

Takeo Kanade

Carnegie Mellon University, Pittsburgh, PA, USA

Josef Kittler

University of Surrey, Guildford, UK

Jon M. Kleinberg

Cornell University, Ithaca, NY, USA

Alfred Kobsa

University of California, Irvine, CA, USA

Friedemann Mattern

ETH Zurich, Switzerland

John C. Mitchell

Stanford University, CA, USA

Moni Naor

Weizmann Institute of Science, Rehovot, Israel

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

University of California, Los Angeles, CA, USA

Doug Tygar

University of California, Berkeley, CA, USA

Gerhard Weikum

Max-Planck Institute of Computer Science, Saarbruecken, Germany

Yan Ma Deokjai Choi Shingo Ata (Eds.)

Challenges for Next Generation Network Operations and Service Management

11th Asia-Pacific Network Operations
and Management Symposium, APNOMS 2008
Beijing, China, October 22-24, 2008
Proceedings

Volume Editors

Yan Ma
Beijing University
of Posts and Telecommunications
Beijing, China
E-mail: mayan@bupt.edu.cn

Deokjai Choi
Chonnam National University
Gwangju, Korea
E-mail: dchoi@chonnam.ac.kr

Shingo Ata
Osaka City University
Osaka, Japan
E-mail: ata@info.eng.osaka-cu.ac.jp

Library of Congress Control Number: Applied for

CR Subject Classification (1998): C.2, B.4, D.2, D.4.4, K.6, H.3.4

LNCS Sublibrary: SL 5 – Computer Communication Networks
and Telecommunications

ISSN 0302-9743
ISBN-10 3-540-88622-2 Springer Berlin Heidelberg New York
ISBN-13 978-3-540-88622-8 Springer 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. Violations are liable to prosecution under the German Copyright Law.

Springer is a part of Springer Science+Business Media
springer.com

© Springer-Verlag Berlin Heidelberg 2008
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12545384 06/3180 5 4 3 2 1 0

Preface

We are delighted to present the proceedings of the 11th Asia-Pacific Network Operations and Management Symposium (APNOMS 2008) which was held in Beijing, China, during October 22–24, 2008.

The Organizing Committee (OC) selected the theme of this year's symposium as "Challenges for Next-Generation Network Operations and Service Management." Research and development on next-generation networks (NGNs) have been carried out over the last few years and we are already seeing their deployment and operations in many parts of Asia-Pacific countries. We are also beginning to experience new and interesting services that utilize these NGNs. We are certain that we will see more deployment of NGNs and NGN services in the next few years. Thus, the operations and management of NGNs and their services are very important to the network operators and service providers. At the same time, they are also concerned about new and more effective ways of performing the operations and management.

This year, the APNOMS call for papers received 195 paper submissions from 19 different countries, including countries outside the Asia-Pacific region (Europe, Middle-East, North and South America). Each paper was carefully reviewed by at least three international experts. Based on review scores, the APNOMS 2008 Technical Program Committee discussed the selection of papers, and selected 43 high-quality papers (22.1% of submissions) as full papers and 34 papers as short papers. Accepted papers were arranged into ten technical sessions and two short paper sessions (poster presentation). These sessions include Routing and Topology Management, Fault Management, Community and Virtual Group Management, Autonomous and Distributed Control, Sensor Network Management, Traffic Identification, QoS Management, Policy and Service Management, Wireless and Mobile Network Management, and Security Management.

The Technical Program Committee (TPC) Co-chairs would like to thank all those authors who contributed to the outstanding APNOMS 2008 technical program. We also thank the TPC, OC members and reviewers for their support throughout the paper review and program organization process. Also, we appreciate KICS KNOM, Korea, and IEICE ICM, Japan, for their sponsorship, as well as IEEE CNOM, IEEE APB, TMF, and IFIP WG 6.6, CIC, CCSA, and BK21 POSTECH FIT for their support for APNOMS 2008.

October 2008

Yan Ma
Deokjai Choi
Shingo Ata

Organization

Organizing Committee

General Co-chairs

Luoming Meng	BUPT, China
James Won-Ki Hong	POSTECH, Korea

Vice Co-chairs

Young-Tak Kim	Yeungnam University, Korea
Hiroshi Uno	NTT, Japan

TPC Co-chairs

Yan Ma, BUPT	China
Deokjai Choi	Chonnam University, Korea
Shingo Ata	Osaka City University, Japan

Tutorial Co-chairs

Peirong Huang	BUPT, China
Hideaki Yamada	KDDI R&D Labs., Japan
Choong Seon Hong	Kyung Hee University, Korea

Special Session Co-chairs

Subin Shen	NUPT, China
Makoto Takano	NTT West, Japan
Won-Kyu Hong	KT, Korea

DEP Co-chairs

Qiliang Zhu	BUPT, China
Yoshiaki Kiriha	NICT, Japan
Kwang-Hui Lee	Changwon National University, Korea

Exhibition Co-chairs

Lianchang Hou	Alcatel-Lucent
Yongchun Liu	Alcatel-Lucent
Tadafumi Ohke	NTT Comware, Japan
Gil-Haeng Lee	ETRI, Korea

Poster Co-chairs

Yongqi He	PKU, China
Naoto Miyauchi	Mitsubishi El., Japan
Young-Seok Lee	CNU, Korea

Publicity Co-chairs

Feng Liu	BJTU, China
Fangnan Yang	BJTU, China
Jun Kitawaki	Hitachi, Japan
Jae-Hyoung Yoo	KT, Korea
Jong-Hwa Park	LG Telecom, Korea
Qinzheng Kong	HP APJ, Australia
Chi-Shih Chao	Feng Chia University, Chinese Taipei

Patrons Co-chairs

Feng Qi	BUPT, China
Young-Myoung Kim	KT, Korea

Finance Co-chairs

Wenjing Li	BUPT, China
Hikaru Seshake	NTT, Japan
Hong-Taek Ju	Keimyung University, Korea

Publication Co-chairs

Jiahai Yang	Tsinghua University, China
Mi-Jung Choi	POSTECH, Korea

Local Arrangements Co-chairs

Weining Wang	BUPT, China
Xiaohong Huang	BUPT, China

Secretaries

Xuesong Qiu	BUPT, China
Ken Masuda	NTT, Japan
Young-Woo Lee	KT, Korea

Advisory Board

Graham Chen	EPAC Tech., Australia
Makoto Yoshida	University of Tokyo, Japan
Masayoshi Ejiri	Studio IT, Japan
Doug Zuckerman	Telcordia, USA
Seong-Beom Kim	KT, Korea

Steering Committee

Nobuo Fujii	NTT, Japan
Hiroshi Kuriyama	NEC, Japan
James W. Hong	POSTECH, Korea
Kyung-Hyu Lee	ETRI, Korea
Young-Tak Kim	Yeungnam University Korea
Yoshiaki Tanaka	Waseda University, Japan

International Liaisons

Ed Pinnes	Elanti Systems, USA
Raouf Boutaba	University of Waterloo, Canada
Carlos Westphall	SCFU, Brazil
Marcus Brunner	NEC Europe, Germany
Rajan Shankaran	Macquarie University, Australia
Alpna J. Doshi	Satyam Computer Services, India
Teerapat Sanguankotchakorn	AIT, Thailand
Borhanuddin Hohd Ali	University of Putra, Malaysia
Victor W.J. Chiu	Chunghwa Telecom, Chinese Taipei
Rocky K. C. Chang	Hong Kong Polytechnic University, China

Technical Program Committee

Co-chairs

Yan Ma	BUPT, China
Deokjai Choi	Chonnam University, Korea
Shingo Ata	Osaka City University, Japan

Members

Adarsh Sethi, University of Delaware, USA	Hyunchul Kim, Seoul National University, Korea
Aiko Pras, University of Twente, The Netherlands	Ian Marshall, Lancaster University, UK
Akira Chugo, Fujitsu Labs, Japan	Iwona Pozniak-Koszalka, Wroclaw University of Technology, Poland
Alexander Keller, IBM, USA	Jae-Oh Lee, University of Technology and Education, Korea
Antonio Liotta, University of Essex, UK	Ji Li, MIT, USA
Carlos Westphall, Federal University of Santa Catarina, Brazil	Jian Gong, Southeast University, China
Cynthia Hood, Illinois Institute of Technology, USA	Jianqiu Zeng, BUPT, China
Filip De Turck, Ghent University, Belgium	Jihwang Yeo, Dartmouth College, USA
Gabi Dreö Rodosek, University of Federal Armed Forces, Munich, Germany	Jilong Wang, Tsinghua University, China
Haci Ali Mantar, Gebze Institute of Technology, Turkey	Jinwoo Kim, Illinois Institute of Technology, USA
Hanan Lutfiyya, University of Western Ontario, Canada	Katsushi Iwashita, Kochi University of Technology, Japan
Haruo Oishi, NTT, Japan	Ki-Hyung Kim, Ajou University, Korea
Hoon Lee, Changwon National University, Korea	Kiyohito Yoshihara, KDDI R&D Labs, Japan
	Kurt Geihs, University of Kassel, Germany
	Lin Zhang, BUPT, China

Lisandro Zambenedetti Granville,
UFRGS, Brazil
Marcus Brunner, NEC Europe Ltd.,
Germany
Masum Hasan, Cisco, USA
Mehmet Ulema, Manhattan College,
USA
Nazim Agoulmine, University of Evry,
France
Olivier Festor, INRIA Nancy - Grand
Est, France
Ping Chen, Peking University, China
Prosper Chemouil, Orange Labs,
France
Rocky K. C. Chang, Hong Kong
Polytechnic University, China
Ruibiao Qiu, F5 Networks Inc., USA
Seongjin Ahn, Sungkyunkwan
University, Korea
Taesang Choi, ETRI, Korea
Teerapat Sanguankotchakorn, AIT,
Thailand

Torsten Braun, University of Bern,
Switzerland
Toshio Tonouchi, NEC, Japan
Xianzhong Xie, Chongqing
University of Posts and
Telecommunications, China
Xiaoyun Zhu, Hewlett Packard Labs,
USA
Yang Cao, Wuhan University, China
Yang Ji, BUPT, China
Yangcheng Huang, Ericsson, Ireland
Yidong Cui, BUPT, China
Yoon-Hee Kim, Sookmyung Women's
University, Korea
Youichi Yamashita, NTT West, Japan
Young Choi, James Madison
University, USA
Zengzhi Li, Xi'an Jiaotong University,
China
Zhiqiang Shi, CAS of China,
China

All OC members are part of the Technical Program Committee as well.

Additional Paper Reviewers

Achilleas Achilleos, University of
Essex, UK
Adetola Oredope, University of Essex,
UK
Alexandre Lefebvre, France Telecom
R&D, France
Anna Sperotto, University of Twente,
The Netherlands
Christophe Dousson, Orange Labs,
France
Clarissa Marquezan, UFRGS, Brazil
Cristiano Both, UFRGS, Brazil
Cristina Melchior, UFRGS, Brazil
Edgar Magana, Cisco, USA
Eric Gourdin, France Telecom
R&D, France
Florence Agboma, University of Essex,
UK

Fu Chen, Tsinghua University, China
Gerald Wagenknecht, University of
Bern, Switzerland
Giorgio Nunzi, NEC Europe Ltd.,
Germany
Guangjie Han, Hohai University, China
Guilherme Sperb Machado, UFRGS,
Brazil
Halim Zaidi, University of Evry, France
Hans-Joerg Kolbe, NEC Europe Ltd.,
Germany
Hui Wang, Tsinghua University, China
Jacek Dzikowski, Illinois Institute of
Technology, USA
Jaehyung Park, Chonnam National
University, Korea
Jitae Shin, Sungkyunkwan University,
Korea

Julien Meuric, Orange Labs, France	Rafael Kunst, UFRGS, Brazil
Kazuhide Takahashi, NTT DoCoMo, Japan	Ramin Sadre, University of Twente, The Netherlands
Kazuhiko Kinoshita, Osaka University, Japan	SangSik Yoon, ETRI, Korea
Kazumitsu Maki, Fujitsu, Japan	Seung-Joon Seok, Kyungnam University, Korea
Kohei Iseda, Fujitsu Labs, Japan	Sumit Naiksatam, Cisco, USA
Majed Alhaisoni, University of Essex, UK	Takafumi Chujo, Fujitsu, Japan
Marat Zhanikeev, Waseda University, Japan	Tetsuya Yamamura, NTT, Japan
Markus Wulff, University of Bern, Switzerland	Thomas Staub, University of Bern, Switzerland
Markus Wehli, University of Bern, Switzerland	Wang-Cheol Song, Cheju National University, Korea
Masaki Aida, Tokyo Metropolitan University, Japan	Weiqiang Sun, Shanghai Jiaotong University, China
Mehdi Nafa, University of Evry, France	Yin Zhenyu, Chinese Academy of Sciences, China
Mylene Pischella, Orange Labs, France	Ying Liu, Cisco, USA
Myung-Sup Kim, Korea University, Korea	Yuichi Ohshita, Osaka University, Japan
Nadia Nawaz Qadri, University of Essex, UK	Yuji Hibino, NTT, Japan
Nguyen Dang, University of Evry, France	Yuka Kato, Advanced Institute of Industrial Technology, Japan
Nobuyuki Takai, Mitsubishi Electric, Japan	Zhang Wenbo, Dalian University, China
	Zheng Hongxing, Dalian Maritime University, China

Table of Contents

Session 1: Routing and Topology Management

E-Scheme in Delay-Tolerant Networks	1
<i>Zhi-Ting Lin, Yu-Gui Qu, Qing-Hua Wang, and Bao-Hua Zhao</i>	
Layer 1 VPN NMS Based on Each VPN Customer	11
<i>Hiroshi Matsuura and Naotaka Morita</i>	
Algorithms to Managing Unicast, Multicast and Broadcast Transmission for Optical Switches	21
<i>Dawid Zydek, Leszek Koszalka, Iwona Pozniak-Koszalka, and Keith J. Burnham</i>	
Network Planning and Adaptive Routing for Multimedia Traffic	31
<i>Priscila Solís Barreto, Paulo H.P. de Carvalho, Rafael Dias Oliveira, and Maximiliano Prestes Ceppo</i>	

Session 2: Fault Management

Network-Wide Rollback Scheme for Fast Recovery from Operator Errors Toward Dependable Network	41
<i>Daisuke Arai, Kiyohito Yoshihara, and Akira Idoue</i>	
An Efficient Failure Recovery Scheme for Next Generation Network Services Based on Distributed Components	51
<i>Wataru Miyazaki, Kazuhiko Kinoshita, Hideki Tode, Koso Murakami, Shinji Kikuchi, Satoshi Tsuchiya, Atsuji Sekiguchi, and Tsuneo Katsuyama</i>	
A Novel Fault Diagnosis Approach to Path-Protected WDM Mesh Networks	61
<i>Chi-Shih Chao</i>	
Active Diagnosis of High-Level Faults in Distributed Internet Services	72
<i>Huihu Long, Lu Cheng, Yongguo Zeng, and Li Wu</i>	
Fault Localization for Heterogeneous Networks Using Alarm Correlation on Consolidated Inventory Database	82
<i>Jinsik Kim, Young-Moon Yang, Sukji Park, Sungwoo Lee, and Byungdeok Chung</i>	

Session 3: Community and Virtual Group Management

Probing Based Topology Inference for Large Scale Community Networks	92
<i>Marat Zhanikeev, Yoshiaki Tanaka, and Tomohiko Ogishi</i>	
Indexing through Querying in Unstructured Peer-to-Peer Overlay Networks	102
<i>K. Haribabu, Chittaranjan Hota, and Antti Ylä-Jääski</i>	
TrustRRep: An Improved Reputation Management Scheme for Reliable Resource Sharing in Peer-to-Peer Networks	112
<i>Junghwa Shin, Taehoon Kim, and Sungwoo Tak</i>	
Group Key Locking Scheme in Mobile Multicast Environments with Modified MLDs	123
<i>Byung-Jin Han, Jong-Hyouk Lee, and Tai-Myoung Chung</i>	

Session 4: Autonomous and Distributed Control

The Best Practice and Evaluation of Processes to Develop OSSs Using XML-Based Interfaces	136
<i>Yuki Kishikawa, Daisuke Fujishima, Hironao Tamura, Kazuhide Takahashi, and Shoichi Hirata</i>	
Review of Knowledge Engineering Requirements for Semantic Reasoning in Autonomic Networks	146
<i>John Strassner, Mícheál Ó Foghlú, Willie Donnelly, Joan Serrat, and Nazim Agoulmine</i>	
Towards Management Requirements of Future Internet	156
<i>Sung-Su Kim, Mi-Jung Choi, Hong-Taek Ju, Masayoshi Ejiri, and James Won-Ki Hong</i>	
Considerations on NETCONF-Based Data Modeling	167
<i>Hui Xu and Debao Xiao</i>	

Session 5: Sensor Network Management

A Hop by Hop Rate Control Based QoS Management for Real Time Traffic in Wireless Sensor Networks	177
<i>Muhammad Mostafa Monowar, Md. Obaidur Rahman, Byung Goo Choi, and Choong Seon Hong</i>	
Extended Concentric-Clustering Routing Scheme Adapted to Various Environments of Sensor Networks	187
<i>Jin-Young Choi, Sung-Min Jung, Young-Ju Han, and Tai-Myoung Chung</i>	

Towards Cluster Based Wireless Sensor Network Deployment Management and Network Coverage Verification	197
<i>Zhanyang Zhang</i>	
A Logical Group Formation and Management Mechanism Using RSSI for Wireless Sensor Networks	207
<i>Jihyuk Heo, Jin Ho Kim, and Choong Seon Hong</i>	

Session 6: Traffic Identification

OD Count Estimation Based on Link Count Data	217
<i>Yi Jin, Dongchen Jiang, Shuai Yuan, Jianting Cao, Lili Wang, and Gang Zhou</i>	
An Efficient Approach for Analyzing Multidimensional Network Traffic	227
<i>Jia Yang, Hao Ma, Bei Zhang, and Ping Chen</i>	
The Measurement of User's Subjective Assessment on Throughput Degradation in Bulk Data Transmission.....	236
<i>Yasuhiro Sato, Shin'ichiro Kitano, Shingo Ata, and Ikuo Oka</i>	
A Space-Efficient Fair Packet Sampling Algorithm	246
<i>Jin Zhang, Xiaona Niu, and Jiangxing Wu</i>	
Understanding IPv6 Usage: Communities and Behaviors	256
<i>Shaojun Huang, Changqing An, Hui Wang, and Jiahai Yang</i>	

Session 7: QoS Management

DTPD: Data Transfer Tool Performance Diagnosis System in High Speed Networks.....	266
<i>Jong-Myoung Kim, Young-Ju Han, Giljae Lee, Woojin Seok, and Tai-Myoung Chung</i>	
QoS-Aware Customer Network Management (Q-CNM) System for Efficient Handovers with PMIPv6 and MIH	276
<i>Young-Chul Jung and Young-Tak Kim</i>	
Dynamic Multi-stream Transport Protocol	287
<i>Seung-Joon Seok, Hyeong-Jun Kim, Kwang-Min Jung, Kyung-Hoe Kim, and Chul-Hee Kang</i>	
A Study on the Service Quality Management Process and Its Realization Strategy for Capturing Customer Value	297
<i>Eunjoo Kwak, Byeong-Yun Chang, Daniel Wonkyu Hong, and Byungdeok Chung</i>	

A Novel Integrated Supporting System for Mesh-Pull Based P2P IPTV	307
<i>Bo Wen, Feng Liu, and Luoming Meng</i>	

Session 8: Policy and Service Management

BPEL Driven Policy Management of Virtualized Network Resources for IMS Environments	317
<i>Nobutaka Matsumoto, Takahiro Miyamoto, Michiaki Hayashi, and Hideaki Tanaka</i>	
Research on the Associated Pricing Strategy in Telecom Industry	327
<i>Xin Yue, Junjie Xu, and Zhanhong Xin</i>	
A Semantic Description Approach for Telecommunications Network Capability Services	334
<i>Xiuquan Qiao, Xiaofeng Li, and Tian You</i>	

Session 9: Wireless and Mobile Network Management

Low Latency Proactive Handover Scheme for Proxy MIPv6 with MIH	344
<i>Igor Kim, Young Chul Jung, and Young-Tak Kim</i>	
Simple Modeling for QoS Management of IEEE 802.11 DCF	354
<i>Takashi Satake</i>	
Take the Challenge of IP Evolution and OAM Solution	363
<i>Lihong Wei and Lingshan Kong</i>	
Performance Evaluation of Heartbeat-Style Failure Detector over Proactive and Reactive Routing Protocols for Mobile Ad Hoc Network	370
<i>Haijun Zhao, Yan Ma, Xiaohong Huang, and Fang Zhao</i>	

Session 10: Security Management

Design and Implementation of an SNMP-Based Traffic Flooding Attack Detection System	380
<i>Jun-Sang Park and Myung-Sup Kim</i>	
A Method to Detect Prefix Hijacking by Using Ping Tests	390
<i>Mitsuho Tahara, Naoki Tateishi, Toshio Oimatsu, and Souhei Majima</i>	
Application of Data Mining to Network Intrusion Detection: Classifier Selection Model	399
<i>Huy Anh Nguyen and Deokjai Choi</i>	

Detection of Leaps/sLumps in Traffic Volume of Internet Backbone	409
<i>Yutaka Hirokawa, Kimihiro Yamamoto, Shigeaki Harada, and Ryoichi Kawahara</i>	

A Sampling Method for Intrusion Detection System	419
<i>Zhuo Ning and Jian Gong</i>	

Session S1, S2: Short Papers

Fast Traffic Classification in High Speed Networks	429
<i>Rentao Gu, Minhao Hong, Hongxiang Wang, Yuefeng Ji</i>	

Service Impact Analysis Framework Using Service Model for Integrated Service Resource Management of NGN Services	433
<i>Seung-Hee Han, Bom-Soo Kim, Chan-Kyou Hwang, and Jae-Jin Lee</i>	

A Trusted Quality of Web Services Management Framework Based on Six Dimensional QoWS Model and End-to-End Monitoring	437
<i>Nan Guo, Tianhan Gao, and Bin Zhang</i>	

Least Interference Optimization Based Dynamic Multi-path Routing Algorithm in ASON	441
<i>Tong Zhao, Yueming Lu, and Yuefeng Ji</i>	

QoS-Aware Scheduling in Emerging Novel Optical Wireless Integrated Networks	445
<i>Min Luo, Hui Li, Yueming Lu, and Yuefeng Ji</i>	

An Energy-Efficiency Route Protocol for MIMO-Based Wireless Sensor Networks	449
<i>Qing-Hua Wang, Yu-Gui Qu, Zhi-Ting Lin, and Bao-Hua Zhao</i>	

Service Oriented T-MPLS Resilience Algorithm with Multi-QoS Constrained	454
<i>Hua Qu, Ji-hong Zhao, and Hong-bao Mao</i>	

Network Stability Analysis Techniques Using the Virtual Articulation Node	458
<i>Yonghyoun Kim, Kisu Kim, Sengjin Ahn, and Jinwook Chung</i>	

A Channel Management Framework to Construct User Preferred Fast Channel Change Stream in IPTV	462
<i>Md. Mamun-Or-Rashid, Dae Sun Kim, and Choong Seon Hong</i>	

A Fair Mobile Payment Protocol	466
<i>Wei Fan, Huaying Shu, Qiang Yan, and Fang Wang</i>	

Statistical Analysis of Slow Portsweep	470
<i>Noriaki Yoshiura</i>	

Empirical Analysis of Application-Level Traffic Classification Using Supervised Machine Learning	474
<i>Byungchul Park, Young J. Won, Mi-Jung Choi, Myung-Sup Kim, and James Won-Ki Hong</i>	
Virtualization-Based Operation Support Systems: Performance Evaluation and Systems Design	478
<i>Yujiro Mochizuki, Kazuhiko Higashi, Kumiko Goto, and Minoru Kato</i>	
A Management Framework for IMS Using Service Managed Objects	483
<i>Muhammad Shoaib Siddiqui, Syed Obaid Amin, and Choong Seon Hong</i>	
Evaluating Open Service Access with an Abstract Model of NGN Functions	487
<i>Samson Lee, John Leaney, Tim O'Neill, and Mark Hunter</i>	
Methods for Rapidly Testing Node Reachability with Congestion Control and Evaluation	491
<i>Naoki Tateishi, Mitsuho Tahara, Yu Miyoshi, and Souhei Majima</i>	
The Design of an Open and Integrated Sensor Network Management Platform	495
<i>Michalis Kalochristianakis, Vasileios Gkamas, Georgios Mylonas, Sotiris Nikolettseas, Jose Rolim, and Emmanouel Varvarigos</i>	
Self-organized Cluster Based Multi-hop Routing for Wireless Sensor Networks	499
<i>Hongjoong Sin, Sungju Lee, Jangsu Lee, Seunghwan Yoo, Sanghyuc Lee, Jaesik Lee, and Sungchun Kim</i>	
An Adaptable Method of E-Workflow Composition Based on Distributed Awareness	503
<i>Hongbin Sun and Yongsheng Ding</i>	
An Adaptive Control Scheme of Reserved Bandwidth for RPR in Steering Mode	507
<i>Wen-Fong Wang, Yi-Shian Chen, and Lih-Chyau Wu</i>	
Estimating Half-Path RTT in Backbone Network	511
<i>Lisheng Huang, Wenyong Wang, and Mingtian Zhou</i>	
A Radio Network Co-design System for Planning, Operation, and Customer Relations Divisions	515
<i>Kosei Kobayashi, Yasuhiko Matsunaga, Takayuki Nyu, and Hiroto Sugahara</i>	
Traffic Matrix Estimation Using Square Root Filtering/Smoothing Algorithm	519
<i>Jingjing Zhou, Jiahai Yang, Yang Yang, and Guanqun Zhang</i>	

A Knowledge-Based Tool to Support Clear Relationship between Threats and Countermeasures Based on International Standards	523
<i>Guillermo Horacio Ramirez Caceres and Yoshimi Teshigawara</i>	
Incentives for Cooperative Relay in Heterogeneous Networks: A Dynamic Reputation-Based Approach	527
<i>Junseok Hwang, Andrei Shin, and Hyenyoung Yoon</i>	
User-Centric Prediction for Battery Lifetime of Mobile Devices	531
<i>Joon-Myung Kang, Chang-Keun Park, Sin-Seok Seo, Mi-Jung Choi, and James Won-Ki Hong</i>	
A Study on the Reliable and Flexible Implementation of Workforce Management Using Business Process Management	535
<i>Kyu-Hwal Kim, Byeong-Yun Jang, Min-kyu Kwon, and Seung-Hak Suk</i>	
Enterprise Management System with Web-Crawler	539
<i>Myung Sil Choi, Yong Soo Park, and Kwang Seon Ahn</i>	
A Construction Process for Small-Scale Network Systems	543
<i>Yuka Kato</i>	
Design, Implementation and Evaluation of a Network Management System for a High Quality IP-Based Video Transmission Service	547
<i>Shuntaro Kashiwara, Kenichi Ogaki, and Tomohiro Otani</i>	
Availability in Peer to Peer Management Networks	552
<i>Ouldooz Baghban Karimi, Saleh Yousefi, Mahmood Fathy, and Mojtaba Mazoochi</i>	
Field Operations Management and Unmanned Operations Management toward NOM 2.0	556
<i>Byeong-Yun Chang, Daniel Wonkyu Hong, Kyu-Hwal Kim, and Byung-Deok Chung</i>	
The IMS/SDP Structure and Implementation of Presence Service	560
<i>Jae-Hyoung Cho and Jae-Oh Lee</i>	
Detection and Handling of TRUNK Congestion in the ATM Network	565
<i>Charlie Yang, Chuan-Chuen Chang, Chi-Ming Chen, David Lu, and Monowar Hossain</i>	
Author Index	569

E-Scheme in Delay-Tolerant Networks

Zhi-Ting Lin^{1,*}, Yu-Gui Qu¹, Qing-Hua Wang¹, and Bao-Hua Zhao²

¹ Dept. of Electronic Engineering and Information Science, ²Dept. of Computer Science,
University of Science and Technology of China, P.O. Box 230027, Anhui, China

² State Key Laboratory of Networking and Switching Technology, P.O. Box 100876 Beijing
ztlin@mail.ustc.edu.cn

Abstract. Delay Tolerant Networks (DTNs) are characterized by very long delay paths and frequent network partitions. In this paper, a novel E-Scheme is proposed which can be utilized by most of the existing routing approaches that employ the ‘probability to deliver’ metric. Providing there are n opportunities, the E-Scheme is to let the first $(n-1)/e$ opportunities go by (except for encountering the destination of messages) and then accept the first one to appear that is better than all of those. Meanwhile a probabilistic metric, called Two-Hop (TH) delivery predictability value, is presented here according to the small world theory. Our scheme has been evaluated using some existing approaches as well as the TH delivery predictability value and been shown to exhibit excellent performances.

Keywords: Delay-Tolerant Network, E-Scheme, routing approach.

1 Introduction

The increasing popularity of devices equipped with wireless network interfaces, such as cell phones or PDAs, offers new communication services opportunities [1]. These Intermittently connected mobile networks fall under Delay Tolerant Networks (DTN) umbrella, where communication contacts are intermittent, an end-to-end path between the source and the destination may have never existed, disconnection and reconnection are common, and link performance is highly changeable [2-6].

Such networks are incurred delays that may be very large and unpredictable, so some well-known assumptions of traditional networks are no longer true in DTNs, which makes routing in DTNs a challenging problem [4].

In this short paper, we focus on the routing issue in the delay tolerant networks and propose a novel common routing scheme. This E-Scheme can be utilized by most of the existing solutions which employ the ‘probability to deliver’ metric. The rest of this paper is organized as follows. In Section 2, we review several ‘probability to

* This paper is supported by the National Natural Science Foundation of China under Grant No. 60602016, the National Grand Fundamental Research 973 Program of China under Grant No.2003CB314801, Hi-Tech Resrarch and Development Program of China 863 No. 2007AA01Z428, HUAWEI Foundation No.YJCB2007061WL and Innovation Fund for Graduate Student of USTC under Grant No. KD2005042.

deliver' metric, analyzing their advantages and the drawbacks. In Section 3, the detail of the common routing e-scheme is presented. And a probabilistic metric called Two-Hop (TH) delivery predictability value is brought forward in Section 4. Simulation results are shown in Section 5, where the performance of E-Scheme is evaluated using some existing approaches as well as the TH delivery predictability value. Section 6 concludes the paper with some discussion on potential open research topics.

2 Existing Probabilistic Metric

The problem of forwarding can be described as: when nodes have a local or global connection opportunity, messages are forwarded according to some policy, with the intention that they are brought closer to their destination [7-10]. So a number of solutions for routing in DTNs have been presented based on recorded encounters where data is routed to nodes with the highest 'probability to deliver' to a destination node [11-13].

2.1 PROPHET

To make use of the observations of the non-randomness of mobility and to improve routing performance, the PROPHET is proposed in Reference 14. When node a encounters b , they exchange a delivery predictability vector. And the calculation of the delivery predictability is shown in Eq.1, where $P_{init} \in (0, 1]$ is an initialization constant.

$$P'_{a,b} = P_{a,b} + (1 - P_{a,b}) \times P_{init} \quad (1)$$

Also, the delivery predictability has a transitive property, which is based on the assumption that messages may be forwarded by relays. ξ is a scaling constant that decides how large impact the transitivity should have on the delivery predictability.

$$P'_{a,c} = P_{a,c} + (1 - P_{a,c}) \times P'_{a,b} \times P'_{b,c} \times \xi \quad (2)$$

Meanwhile, the delivery predictability values must age, being reduced in the process. The aging equation is shown in Eq.3, where $\gamma \in (0, 1)$ is the aging constant, and τ is the number of time units that have lapsed since the last time the metric was aged.

$$P'_{a,b} = P_{a,b} \times \gamma^\tau \quad (3)$$

2.2 MRN

As mentioned before, the key to the routing strategy is the computation of the next hop on the path to the destination [12, 15]. Since the history is often a good predictor for the future movement of physically mobile components. It is reasonable to suppose that a host which has recently noticed the target will do so again in the future. Therefore the probability assigns a higher value for a node which has recently encountered the destination:

$$P_{MRN} = (1 - \frac{CurrentTime - LastTimeNotice_D}{TimeOut_m}) \times 100\% \quad (4)$$

where $LastTimeNoticed_D$ is the time stamp for latest noticing D , $CurrentTime$ is the current wall clock time, and $TimeOut_m$ serves to normalize the MRN utility among all hosts.

2.3 MFN

By adding a little overhead to the MRN method, we can also record information relevant to the frequency of noticing a host [15], i.e. the number of encounters and the time of the first encounter. By means of this info, probability can be calculated as:

$$P_{MFN} = (1 - \frac{CurrentTime - FirstTimeNotice_D}{TimeOut_m \times NumTimesNoticed_D}) \times 100\% \quad (5)$$

3 E-Scheme

In DTNs, nodes should store the packets and upon each encounter with another node, the decision must be made on whether or not to transmit a particular message. However encounters, i.e. transmission opportunities, are presented in sequence. After being presented with a delivery predictability value, node has to immediately determine whether to accept it or not. For most of the existing routing approaches, messages are promptly transmitted to nodes that have higher delivery predictability values. However our intuition is that it might be wise to wait till a much better opportunity to minimize the number of relays. But how and when should nodes transfer messages to others. The only information we are given when deciding whether to accept the opportunity is the ones already rejected [16]. That is, for instance, when the fifth opportunity is presented, we learn how it compares with the first four opportunities already seen. Suppose that once an opportunity is rejected it is lost, and that our intention is to maximize the probability of obtaining the best opportunity.

Here, we assume that a message must be retained in the buffer before rejecting n opportunities. In E-Scheme, the first k opportunities will be rejected and then the first one which is better than all of those first k will be accepted. Although that will be up to $n!$ orderings of the opportunities are equally likely, E-Scheme can settle it quite well. Let P_k denote the probability that the best opportunity is selected when E-Scheme is employed. To compute this probability, we can condition on X , the position of the best opportunity.

$$P_k(best) = \sum_{i=1}^n P_k(best | X = i)P(X = i) = n^{-1} \sum_{i=1}^n P_k(best | X = i) \quad (6)$$

Now, if the overall best opportunity is among the first k , then no opportunity is ever selected. However if the best opportunity is in position i , where $i > k$, then the best opportunity will be select if and only if the best of the first k opportunities is also the best of the first $i-1$ opportunities. Because under such condition, none of the opportunities in positions $k+1, k+2, \dots, i-1$ would be chosen. Therefore, we can induce that

$$\begin{aligned} P_k(\text{best} | X = i) &= 0 & \text{if } i \leq k \\ P_k(\text{best} | X = i) &= k/(i-1) & \text{if } i > k \end{aligned} \quad (7)$$

From Eq(6) and (7), we can infer that

$$P_k(\text{best}) = \frac{k}{n} \sum_{i=k+1}^n \frac{1}{i-1} \approx \frac{k}{n} \int_k^{n-1} \frac{1}{x} dx = \frac{k}{n} \log\left(\frac{n-1}{k}\right) \quad (8)$$

Now if we consider the function

$$P_k'(\text{best}) = \frac{1}{n} \log\left(\frac{n-1}{k}\right) - \frac{1}{n} \quad (9)$$

And so

$$P_k'(\text{best}) = 0 \Rightarrow \log\left(\frac{n-1}{k}\right) = 1 \Rightarrow k = \frac{n-1}{e} \quad (10)$$

In other words, the E-Scheme is to let the first $(n-1)/e$ opportunities go by (except for encountering the destination of messages) and then accept the first one to appear that is better than all of those. After n opportunities, the message won't be guaranteed to be retained in the buffer any more. In addition, since $P_{(n-1)/e}(\text{best}) = 1/e$, the probability that E-Scheme selects the best opportunity is over $1/e \approx 0.36788$. Obviously, E-Scheme can be used in the single-copy approaches to increase the delivery ratio, i.e. the amount of packets received at destination.

Furthermore, the E-Scheme can be considered as a class of Markov chains, known as branching processes [16]. Suppose that each message will produce a new offspring with probability $P_{\text{offspring}_1} \approx (n-k)/n$, independently of the numbers produced by other messages. The number of messages initially present, denoted by X_0 , is called the size of the 0th generation. All offspring of the 0th generation constitute the first generation and their number is denoted by X_1 . In general, let X_n denote the size of the n^{th} generation. It follows that $\{X_n, n=0,1,\dots\}$ is a Markov chain having as its state space the set of nonnegative integers. Let

$$\mu = 1 \times P_{\text{offspring}_1} + 0 \times P_{\text{offspring}_0} \quad (11)$$

denote the mean number of offspring of a single message, and let

$$\sigma^2 = (1-\mu)^2 \times P_{\text{offspring}_1} + (0-\mu)^2 \times P_{\text{offspring}_0} \quad (12)$$

be the variance of the number of offspring produced by a single message. Let us suppose that $X_0=q$, that is, initially there are q messages in the delay tolerant networking. $E[X_n]$ and $\text{Var}[X_n]$ can be calculated as

$$E[X_n] = E[E[X_n | X_{n-1}]] = E\left[E\left[\sum_{i=1}^{X_{n-1}} Z_i | X_{n-1}\right]\right] = uE[X_{n-1}] = q\mu^n \quad (13)$$

$$\text{Var}(X_n) = E[\text{Var}(X_n | X_{n-1})] + \text{Var}(E[X_n | X_{n-1}]) = \sigma^2(\mu^{n-1} + \mu^n + \dots + \mu^{2n-2}) \quad (14)$$

where Z_i represents the number of offspring of the i^{th} message of the $(n-1)^{\text{st}}$ generation, and the fact that $E[Z_i] = \mu$ has been used.

Although the E-Scheme would be shown to exhibit excellent performance in the following simulation section, according to the Eq.13 and Eq.14 the amount of messages will decrease when they are transmitted to other nodes. In order to resolve this drawback, E-Scheme can be revised as utilizing multi-copy: (1) rejecting first $(n-1)/e$ opportunities (except for encountering the destination of messages) and then accepting multiple opportunities which appear better probability than the first $(n-1)/e$ ones; or (2) accepting the first opportunity and then utilizing the original E-Scheme.

4 Two-Hop Delivery Predictability

The increasing popularity of devices equipped with wireless network interfaces offers new communication services opportunities. Node movement is often affected by human decisions and socialization behaviors. In terms of sociology theory, the small-world phenomenon formalizes the anecdotal notion that one is only ever six degrees of separation away from anybody else. And it's well known that in the small world [17], everyone seems to be at the centre, because everyone is close to everyone else. Take the Kevin Bacon Game for instance. Over 85% actors could be connected to Kevin Bacon in only three hops. So in this paper, we propose the Two-Hop Delivery Predictability Value according to the small world theory. To accomplish this, we establish a neighbor list with $\sqrt{\Omega}$ items at every node, where Ω denotes the number of nodes in the network. The neighbor list is composed of the IDs, the times of encounter, the time stamp for each one-hop neighbors and the neighbors of them. The structure of the neighbor list is illustrated as Fig.1.

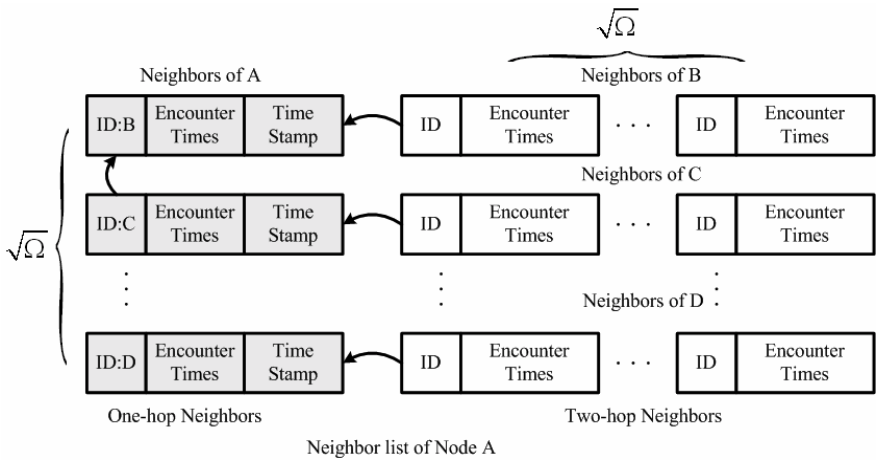


Fig. 1. The structure of the neighbor list in the Two-Hop delivery predictability value

Whenever two nodes encounter, they are required to switch their one-hop neighbors' list. For instance, node A should notify that node B, C, ..., D are one-hop neighbors of it. And if node Z receives this notice, firstly it will go through the local neighbor list. If A is already recorded in one-hop neighbors' list, then update this item by adding 1 to the encounter times, recording the encounter time, and refreshing the two-hop information. Otherwise, add a new item in the one-hop neighbors' list if the amount of one-hop neighbors has not exceeded $\sqrt{\Omega}$. Further more, if node Z has already kept $\sqrt{\Omega}$ items, then find out and replace the obsolete item according to $\text{MAX}[(\text{CurrentTime} - \text{TimeStamp}) / \text{EncounterTimes}]$. By switching the neighbor information, nodes can calculate delivery predictability value for each message stored in buffer. For example, if the destination of the message is node i , then the delivery predictability value can be calculated as:

$$TH(i) = \begin{cases} \log(i.\text{EncounterTimes} + 1) & \text{if } i \in \{\text{One-hop Neighbors}\} \\ 0.5 * \log \sqrt{(i.\text{EncounterTimes} + 1) \times (j.\text{EncounterTimes} + 1)} & \text{if } j \in \{\text{One-hop Neighbors}\} \\ & \& i \in \{\text{Neighbors of } j\} \\ 0 & \text{others} \end{cases} \quad (15)$$

This indicates how likely it is that this node will be able to deliver a message to that destination. And nodes can obtain clue to the destination as far as 3-hop away by switching TH delivery predictability and determine whether to transfer messages or not.

5 Simulations

It's observed that mobile nodes in realistic DTNs may always have some mobility pattern information which can be used to estimate node's ability to deliver a specific message [2]. With this in mind, we develop a wireless mobile model called Wireless Social Model (WSM) to evaluate the E-Scheme.

5.1 Mobility Model

In order to model the existing relationships in the real world, it's necessary to figure out how to generate nodes' connections. Fortunately, in recent years, social networks have been investigated in considerable details [17]. The construction of α -Model is as follows:

$$R_{i,j} = \begin{cases} 1, & m_{i,j} > k\_degree \\ \left[\frac{m_{i,j}}{k\_degree} \right]^\alpha (1 - p) + p, & k\_degree > m_{i,j} > 0 \\ p, & m_{i,j} = 0 \end{cases} \quad (16)$$

In the language of graph theory [17], a DTN consists of a nonempty set of elements, called vertices, and a list of unordered pairs of these elements, called edges. $R_{i,j}$ denotes a measure of the vertex i 's propensity to connect to vertex j . Specially, $R_{i,j} = 0$, if a connection has existed between them. In other words, the possibility of

reconnection is zero. $m_{i,j}$ stands for the number of common adjacent vertices of i and j . k_degree equals to the average degree of the graph. Finally, α and p are tunable parameters, where p is the random probability of an edge (i,j) .

Nodes in Wireless Social Model are driven by the relationship based on α -Model and have an equal probability to move to one of his adjacent vertices. More precisely, a node begins by staying in one location for a random period of time uniformly distributed between $[min\_period, max\_period]$. Once this time expires, it chooses its new goal according to (17). In other words, when the mobile nodes reach their goal they may stay for a while with a possibility of β , wander randomly with a possibility of δ , or head for another one-hop friend. Nodes encounter each other more frequently if they have a social connection.

$$G(i) = \begin{cases} G(i) , & p < \beta \\ Random, & \beta + \delta > p \geq \beta \\ G(j), & \frac{j(1-\beta-\delta)}{k} + \beta + \delta > p \geq \frac{(j-1)(1-\beta-\delta)}{k} + \beta + \delta \end{cases} \quad (17)$$

5.2 Evaluation

Here, we assume that every node has a finite buffer size so that messages may be discarded because of buffer overflow. Two nodes transfer data packets to each other when within communication range. During a transfer, the sender won't replicate packets. A node can deliver packets to a destination node directly or via intermediate nodes, but packets may not be fragmented.

The default settings of the network of our simulator in NS2 are listed in Table 1, using the values recommended by [14].

Table 1. Parameter Settings

Parameter	Default Value	Parameter	Default Value
Num of nodes	100	Stay Period	1500-2500s
Area	5000*5000m ²	β	0.4
Duration	500 000s	δ	0.1
MAC	802.11	P_{init}	0.75
Bandwidth	54Mb/s	ξ	0.25
CommunicationRange	20m	γ	0.98
Speed	2~6m/s	Predictability Buffer(TH)	10
Replication	Never	Predictability Buffer(others)	100
k_degree	6	Total Opportunities n	11
α	2	Skip Opportunities k	4
p	$5*10^{-5}$	Packets/buffer	70~700

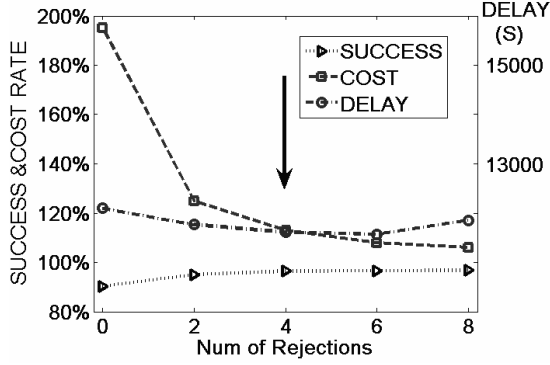


Fig. 2. Performance of E-Scheme using TH delivery predictability (vary num of rejections)

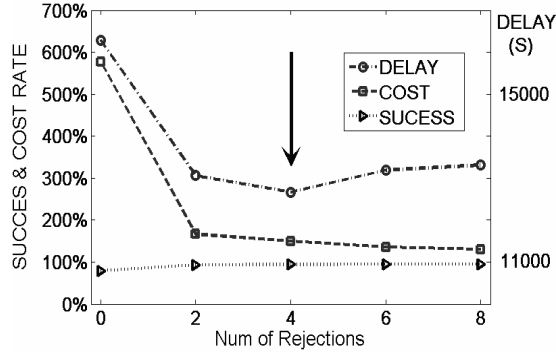


Fig. 3. Performance of E-Scheme using ProPHET delivery predictability (vary num of rejections)

Firstly, we vary the number of rejections from 0 to 8 so as to find out the best k . Success rate of delivery, cost rate and the delay of messages are plotted in Fig.2 and Fig.3, where cost rate denotes the *total packets/(Message generation rate* Simulation time)* in the network. According to the Eq.10 the best value of k is $(11-1)/e=3.66$ which matches the simulation results. Here, we set the *Message generation rate* Simulation time/buffer* to 70. Results for other value (140, 700) are not shown due to lack of space, but generally show the similar behavior. The results in Fig.3 show that routing with E-Scheme outperforms original ProPHET in terms of Success rate of delivery and cost rate by at least 75%.

Our intuition is that the average delay may increase by introducing E-Scheme. Surprisingly the average delay of messages decreases simultaneously. It's because that messages are kept in buffer till a much better opportunity.

Figure 4 shows a CDF of the message delivery delays for all the four predictabilities. So as to take a closer look at the CDF, we only present a portion from 0.4 to 0.7, which illustrates that routing packet with E-Scheme has shorter delays.

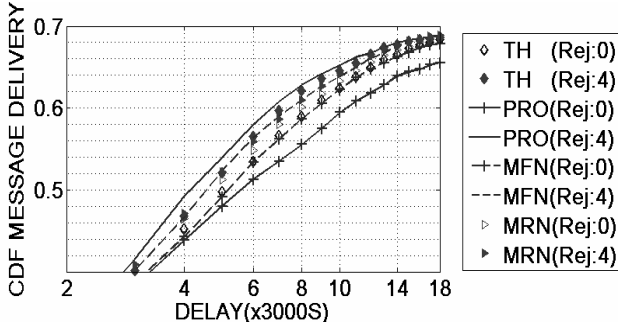


Fig. 4. CDF of delay of messages for TH, PROPHET, MFN and MRN

6 Conclusion

DTN routing appears to be a rich and challenging problem. In this work we propose a simple but effective approach, called E-Scheme, which manages to overcome the shortcomings of existing predictability-based schemes. Providing there are n opportunities, the E-Scheme is to let the first $(n-1)/e$ opportunities go by (except for encountering the destination of messages) and then accept the first one to appear that is better than all of those. Our method has been compared with four different predictabilities value via simulations and been shown to exhibit excellent performance.

References

1. Chaintreau, A., Jui, P., Crowcroft, J., Diot, C., Gass, R., Scott, J.: Impact of human mobility on the design of opportunistic forwarding algorithms. In: Proceedings of IEEE INFOCOM, Barcelona, ES (2006)
2. Song, L., Kotz, D.: Evaluating opportunistic routing protocols with large realistic contact traces. In: ACM MobiCom workshop on Challenged Networks (2007)
3. Chen, L.-J., Yu, C.-H., Sun, T., Chen, Y.-C.: A hybrid routing approach for opportunistic networks. In: CHANTS 2006: Proceedings of the 2006 SIGCOMM workshop on Challenged networks, pp. 213–220 (2006)
4. Elizabeth, D., Mads, H.: Social Network Analysis for Routing in Disconnected Delay-Tolerant MANETs. In: MobiHoc 2007, pp. 32–40 (2007)
5. Muhammad, A., Robert, S.: The Impact of the Mobility Model on Delay Tolerant Networking Performance Analysis. In: Proceedings of the 40th Annual Simulation Symposium (2007)
6. Vahdat, A., Becker, D.: Epidemic routing for partially-connected ad hoc networks. Technical Report CS-2000-06, Duke University (2000)
7. Hui, P., Crowcroft, J.: Bubble rap: Forwarding insmall world dtns in ever decreasing circles. Technical Report UCAM-CL-TR-684, University of Cambridge (2007)
8. Fall, K.: A Delay-Tolerant Network Architecture for Challenged Internets. SIGCOMM (2003)
9. Lindgren, A., Doria, A.: Probabilistic routing protocol for intermittently connected networks. Technical report, draft-lindgren-dtnrg-prophet-01.txt, IETF Internet draft (2005)

10. Lindgren, A., Doria, A., Schelen, O.: Probabilistic routing in intermittently connected networks. *ACM SIGMOBILE Mobile Computing and Communications Review* 7(3), 19–20 (2003)
11. Burgess, J., Gallagher, B., Jensen, D., Levine, B.N.: Maxprop: Routing for vehicle-based disruption-tolerant networking. In: *Proc. Infocom 2006*, vol. 4, pp. 1688–1698. IEEE, Los Alamitos (2006)
12. Dubois-Ferriere, H., GrossGlauser, M., And Verrerli, M.: Age matters: efficient route discovery in mobile ad hoc networks using encounter ages. In: *Proc. MobiHoc 2003*, pp. 257–266. ACM Press, New York (2003)
13. Musolesi, M., Hailes, S., Masxolo, C.: Adaptive routing for intermittently connected mobile ad hoc networks. In: *Proc. WoWMoM 2005*, pp. 183–189. IEEE, Los Alamitos (2005)
14. Lindgren, A., Doria, A., Schelén, O.: Probabilistic routing in intermittently connected networks. In: *SIGMOBILE Mobile Computing Communications Review* (2003)
15. Chen, X., Murphy, A.L.: Enabling Disconnected Transitive Communication in Mobile Adhoc Networks. In: *Workshop on Principles of Mobile Computing* (August 2001)
16. Ross, S.M.: *Introduction to Probability Models*. 9. Elsevier, Amsterdam (2007)
17. Watts, D.J.: *Small Worlds.: the dynamics of networks between order and randomness*. Princeton University Press, Chichester (1999)

Layer 1 VPN NMS Based on Each VPN Customer

Hiroshi Matsuura and Naotaka Morita

NTT Service Integration Laboratories

9-11, Midori-Cho 3-Chome, Musashino-Shi, Tokyo 180-8585, Japan

matsuura.hiroshi@lab.ntt.co.jp, morita.naotaka@lab.ntt.co.jp

Abstract. A virtual private network (VPN) makes it possible for a customer to use a wide range network by borrowing a provider network. In addition, by recent emergence of the layer 1 VPN (L1VPN), more customers require to control the end-to-end path setting on VPN by using path computation traffic engineering. We propose a customer-based L1VPN network management system (NMS) that is separated from a provider NMS. This customized NMS enables each customer to explicitly control a customer IP path from an end-to-end point of view. The separation of each customer L1VPN NMS from a provider NMS also releases the provider NMS from a huge burden. We implement a management and control function in an L1VPN NMS by using enterprise java beans (EJBs). Then, we discuss and evaluate the merits of the proposed NMS.

Keywords: L1VPN, GMPLS, PCE, EJB, NMS.

1 Introduction

The layer 1 virtual private network (L1VPN) [1] is gaining attention due to global use of optical networks and flexible setting of lambda optical paths. Instead of using lambda paths, other L1 paths such as the TDM path can be an alternative in the L1VPN. However, we deal with a lambda path as an L1 path in this paper because of the growing use of optical fiber networks. The L1VPN makes it possible for an IP customer network domain to be connected to another IP customer network domain via a lambda path, which is set between the domains. The lambda paths are provided by a network provider based on the contract between a customer and the provider in terms of bandwidth among customer domains and quality of service (QoS).

Generalized multiprotocol label switching (GMPLS) [2] is applied to connect IP customer domains with an optical provider network, so that end-to-end paths are set on this multilayered L1VPN network. So far, various traffic engineering algorithms [3] [4] on GMPLS networks have been proposed in multi-layered environments. The path computation element (PCE) [5] is standardized to calculate the appropriate end-to-end path route on a GMPLS network, and this PCE solution is also considered to be applied to the L1VPN because L1VPN consists of GMPLS multilayered networks, namely IP and optical networks.

In an L1VPN there are customer edge (CE) routers and provider edge routers. Between two CEs, lambda paths are established to carry customer traffic between them.

There are two models in the L1VPN, in terms of the usage of these lambda paths: one of them is a dedicated model, and the other is a shared model. In the dedicated model, lambdas are fixedly allocated to each customer. On the other hand, in the shared model, a lambda is created when it is necessary and deleted when it is no longer necessary. In both models, effective resource usage in the L1VPN is discussed [6] [7]. Appropriate routes of lambda paths are calculated by using PCEs, and we proposed the calculation algorithm [6] which considers end-to-end link cost even in customer domains.

However, in a customized L1VPN management, there is some room left for research and experimentation. For example, [8] shows the relationship between control and management planes for the L1VPN. However, there are no clear management operations shown, and there is no explanation of how to separate each L1VPN network management system (NMS) for each customer from a provider NMS. [9] emphasizes a customized NMS for each L1VPN. However, the management information sent from the provider NMS to the L1VPN NMS is supposed to be manually given. In addition, [9] only deals with the L1VPN configuration among CEs and does not consider the customer domain IP links. Once there are two or more CEs in one customer domain, considering IP links is indispensable to choose the best end-to-end route.

We propose an L1VPN NMS that is separated from a provider NMS and customized for each VPN customer. The basic management information in each L1VPN is created by the provider NMS and sent to the L1VPN NMS automatically. After that, a customer operator can control the allocated network resources based on the customer policy. In addition, the L1VPN NMS receives the virtual links, which refers to the logical links between two routers in a customer domain, from a customer domain NMS. Therefore, the end-to-end view, which consists of customer domains and a provider network, can be considered.

Our L1VPN NMS also conducts the configuration and fault management for each customer. For example, we can pinpoint the effect on a user once a network problem occurs in a provider network. Sometimes management functions and control functions badly affect each other and degrade the performance of the NMS. However, we separate enterprise java beans (EJB) Entities [10] used for a control function from those for a management function, so that the effect of each function on the other is minimized.

2 NMS Architecture

We summarize three types of NMS: provider NMS, customer domain NMS, and an L1VPN NMS. After explaining each role of the NMS, we show the fault management architecture in which those three NMSs cooperate. We also show the effect of separating the L1VPN NMS for each customer by measuring the processing time.

2.1 Each NMS's Role

An example of management targets for each NMS is shown in Fig. 1. The provider NMS and customer domain NMS were already proposed to manage a corresponding provider network or customer domain, and to cooperatively determine the interdomain routes hierarchically [6]. In this paper, we propose an L1VPN NMS, which lightens the burden of the provider NMS and enables a customized view of control and management for each VPN customer.

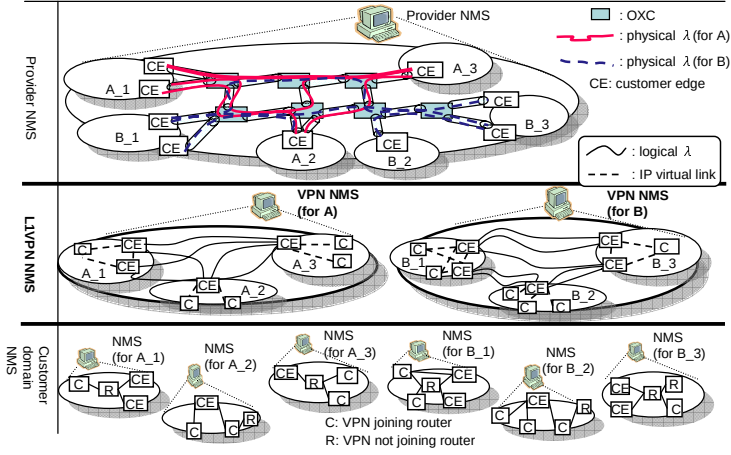


Fig. 1. L1VPN NMS and related provider NMS and customer domain NMS

A provider NMS manages an optical network among customer IP domains, thus the provider NMS has EJB Entities [10] for optical fibers and optical cross connectors (OXCs) and customer domains as managed objects (MOs). An EJB Entity corresponds to a DB table. Based on the contract with a customer, a provider NMS creates lambda paths based on the shortest routes between the specified two customer domains or CEs. After creating those physical lambda paths as Entities in a provider NMS, the provider NMS asks the L1VPN NMS to create logical lambda Entities corresponding to the physical lambda Entities.

A customer domain NMS manages IP routers including CEs and IP links between them as Entities. It also manages IP connections, which are created in IP links. A customer domain NMS discloses routers named Cs that participate in the L1VPN to L1VPN NMS, and it also periodically renews virtual links and discloses them to an L1VPN NMS. A virtual link is a logical link between a C and CEs and has an available bandwidth and its cost as its attributes, and the available bandwidth is defined as the bandwidth of the smallest available bandwidth of its component IP links.

The role of a logical lambda Entity in an L1VPN NMS is quite different from the role of a physical lambda Entity in a provider NMS. A physical lambda is used for the configuration and fault management of the lambda path. Thus, it has component OXCs and optical fibers as its attributes. On the other hand, a logical lambda is used for routing purposes in the L1VPN NMS because a lambda path in an L1VPN becomes one of the IP links for IP paths. As well as a virtual link, a logical lambda has available bandwidth and cost as its attributes. Therefore, virtual links and logical lambdas contribute to select an IP route, which has the cheapest cost and sufficient bandwidth from source IP router to sink IP router. For example, if there is an offer from the source router in the A_1 customer domain to setup the route from the C in A_1 to one of the Cs in A_3, the NMS for the A_1 customer domain sends a message to the L1VPN NMS for customer A to determine the interdomain route. The L1VPN NMS chooses the cheapest route between the specified Cs using the Dijkstra algorithm [11]. After determining the route, the L1VPN NMS asks the NMSs for A_1

and A_3 to determine the detailed route from the source C to CE in A_1 and CE to C in A_3 simultaneously. The cost of logical lambdas and virtual links are automatically updated once the available bandwidth is changed because we choose the least-loaded (LL) algorithm [12], which sets the cost of a link as the inverse of its available bandwidth.

The L1VPN NMS is dedicated to each customer, thus manually changing the cost of the logical lambda path by the L1VPN NMS operator is possible. For example, the operator can increase the cost of low performance lambda until the problem is solved. This kind of manipulation of the cost of logical lambda was not available in our previous work [6].

2.2 Fault Management Architecture

Previous L1VPN-related works [6] [7] [8] [9] did not mention the fault management architecture, let alone the method of how to cooperate among different types of NMSs to handle faults. In our proposal, provider network failures and customer network failures are sent to the L1VPN NMS through the alarm notifications from both provider and customer domain NMSs.

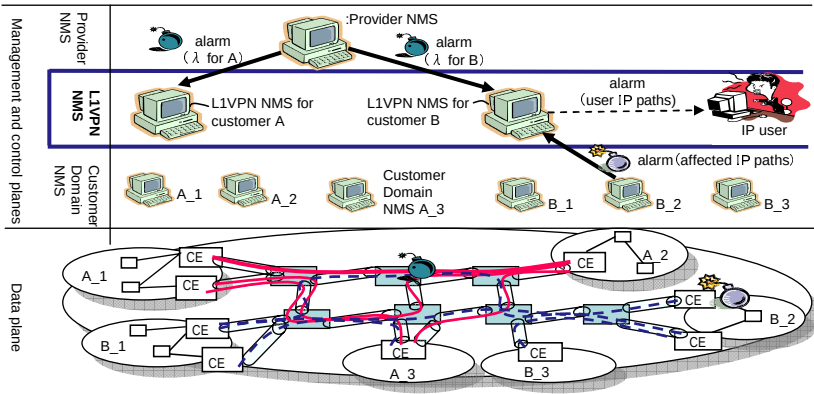


Fig. 2. Fault management by cooperation of three NMSs

Examples of fault management by cooperation of three NMSs are shown in Fig. 2. In Fig. 2, in the provider network an OXC port for an optical fiber is supposed to have a failure and lambda paths in the fiber become no longer available. The provider NMS can identify an affected logical lambda Entity because each physical lambda Entity has its corresponding logical lambda Entity identifier (ID). After the identification of these affected logical lambda Entities, the provider NMS identifies the customers who own these lambdas and send alarms to the corresponding L1VPN NMS attaching logical lambda Entity IDs. A provider NMS can identify the customer because each physical lambda Entity has its owner's customer ID. Each L1VPN NMS can identify IP paths in the specified logical lambda paths and pinpoint users who are using these IP paths. If necessary, the L1VPN NMS sends a notification that includes cause of failure, prospective recovery time, and countermeasure, for example.

In Fig. 2, in customer domain B_2 an IP link between C and CE is supposed to have a failure and IP connections between them are no longer available. IP connection Entities are assigned their IP path Entity ID when an IP connection is created as a component of the IP path. Therefore, a customer domain NMS can identify all affected IP paths and send the list of IP path Entity IDs to the L1VPN NMS for customer B.

Without separating each L1VPN NMS from a provider NMS, it is difficult for the provider NMS to change the policy of dealing with failure alarms depending on IP path users because the provider NMS has to have many customers and does not set the policy for each IP path user. In addition, the provider NMS is estimated to have a large burden to deal with, so there are many customer-related alarms without the help of L1VPN NMSs.

2.3 Evaluation: Effect of L1VPN NMS Separation from Provider NMS

One of our two proposals in this paper is allocating an L1VPN NMS to each customer. Besides the customized control and management performed by each customer operator, the performance of each operation is considered to be faster because of distributing the burden to each L1VPN NMS. The evaluation environment for an operational processing time is shown in Fig. 3. We use four HP ProLiant servers [13] with the RedHat OS installed. We used JBoss 4.2.0 GA application server [14] which follows EJB3 specifications [10]. We used the MySQL Community Edition 5.0.51a database [15] in each server. Four servers are connected by a 1-Gbps LAN. One of the four servers has a provider NMS in it, and the other three servers have an L1VPN NMS in each of them. Each NMS consists of two processes: a JBoss client and a JBoss server. A JBoss client requests operations from its corresponding JBoss server through a message-driven bean (MDB) [10], which enables handling an asynchronous request from the client.

When the provider NMS asks L1VPN NMSs to perform operations, the provider NMS directly sends requests to the MDBs without passing through JBoss clients in L1VPN NMSs. When the provider MDB sends requests to L1VPN MDBs the queues

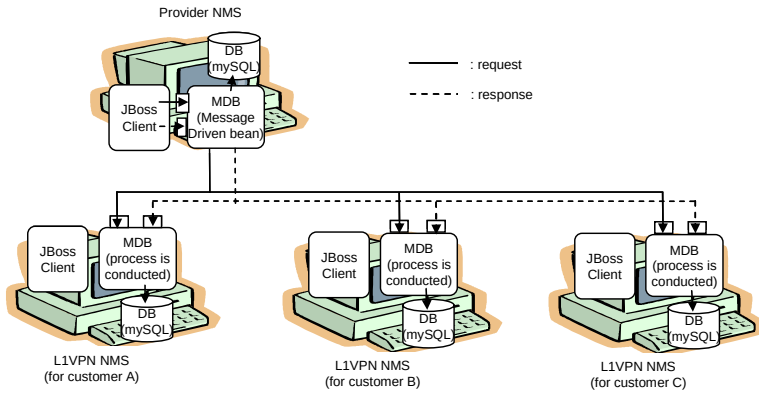


Fig. 3. Evaluation environment

of L1VPN NMSs are used. When the provider MDB receives responses from L1VPN MDBs, another queue of each L1VPN NMS is monitored by the provider MDB to wait for the response. Without the help of L1VPN NMSs, the provider NMS has to deal with all L1VPN-related operations. Therefore, we compare the processing time between two different conditions: one condition requires the provider NMS to conduct all three customer-related jobs, and the other condition requires the provider NMS to ask three underlying L1VPN NMSs to conduct each customer-related job.

The result of this evaluation and approximate functions that indicate the relationship between the processing time for each customer-related job and the entire processing time are shown in Fig. 4. As shown in the graph, we can see the measured plots approximately follow the functions. We estimate the function for the relationship between the processing time for each customer-related job and entire processing time as $y = 206.6 + 3x$, if the provider NMS conducts all three jobs. This is because in the ten plots, the average processing time except for three customer-related jobs is 206.6 ms, and the provider NMS cannot parallelize 3 jobs in which each processing time is x . On the other hand, when the provider NMS delegates the three jobs to corresponding L1VPN NMSs, the estimated function is $y = 576.4 + x$. This is because the average processing time except for the processing time for each customer-related job is 576.4 ms, and three customer-related jobs, in which each processing time is x , are conducted simultaneously in three different L1VPN NMSs.

As shown in the graph, the parallel processing by three L1VPN NMSs is effective, especially when the job in each L1VPN NMS takes a long processing time. From the estimated functions, we can say that as the number of L1VPN NMSs increases, more jobs are parallelized and the whole processing time becomes relatively shorter.

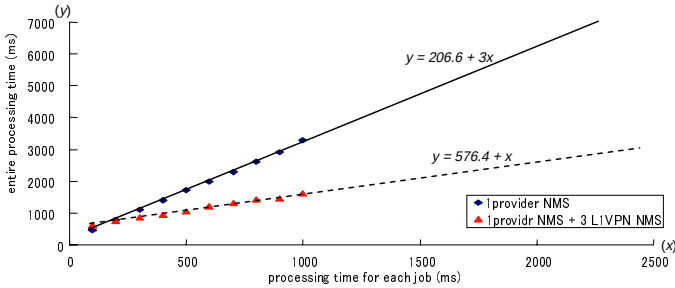


Fig. 4. Measured values and functions gained from values

3 Functional Architecture

ITU-T M.3100 [16] defines managed objects (MOs) for a generic network information model, and ITU-T G.805 [17] defines MOs for generic transport networks. However, there are no documents that define the specific MOs for the L1VPN. We propose MOs for control and management purposes separately in each NMS, and implement these MOs by Entities [10] in EJB. We show the functional architecture in each NMS and evaluate the performance.

3.1 Separation of Management Entity from Control Entity

The main Entities held in a provider NMS are shown in Fig. 5. The provider NMS has CEs, OXCs, and optical fibers between these network elements (NEs), as the control Entities. However, optical fibers between the same NE pair are dealt with as one “control fiber”, which means the optical fiber Entity for a control purpose. These three types of Entities are used for calculating the best route for the lambda path, whose terminated customer domains or CEs are specified. The interdomain SPF algorithm [3] is used to calculate the shortest path among customer domains. Therefore, a “control fiber” has fiber cost and available bandwidth. CE and OXC Entities have an attribute “control fibers” that start from these NEs.

On the other hand, as management Entities, the provider NMS also has CEs, OXCs, and optical fibers. A “management fiber”, which is different from the “control fiber”, is the optical fiber Entity for a management purpose that corresponds to each physical optical fiber. A “management fiber” has a list of lambda paths in it as its attribute, and this attribute is used for the purpose of configuration management. CE and OXC management Entities have management information such as boards and ports in them. As shown in Fig. 5, even though control Entities cover the same NEs and fibers in the data plane as management Entities do, the roles of control Entities are different from those of management Entities.

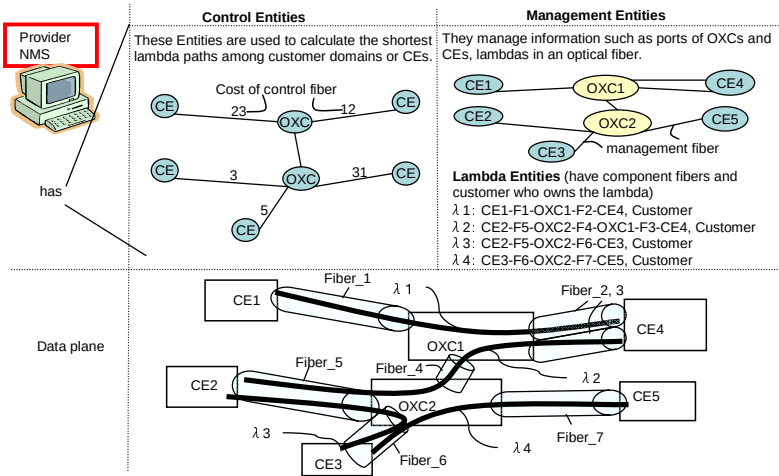


Fig. 5. Main control Entities and management Entities in provider NMS

L1VPN NMS, and customer domain NMS, also have their own control Entities and management Entities. For example in an L1VPN NMS, a logical lambda Entity for control is used for calculating the shortest VPN IP route from one customer domain to the other, whereas a logical lambda Entity for management is used for configuration management of IP paths in the lambda path. In a customer domain NMS, an IP link Entity for control is used for calculating shortest IP routes in the domain to update the route-list, whereas an IP link Entity for management is used for configuration management of IP connections in the IP link.

The common EJB architecture in a JBoss server for provider, L1VPN, and customer domain NMSs to handle control and management requests from a JBoss client is shown in Fig. 6. MDBs and Entities are separated based on functions: control function and management function. As an example of management operation in a provider NMS, an MDB receives a request to return a list of lambda paths in a fiber by specifying the fiber. As one of the control operations, an MDB receives a request to create a list of lambdas for a special customer by specifying customer domains and required bandwidth among them. Generally, a control operation takes a longer processing time than that of a management operation. Therefore, it is important that management operations receive the minimum effect from control operations.

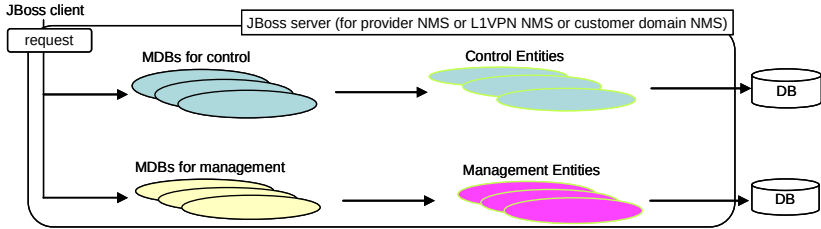


Fig. 6. EJB architecture in JBoss server

3.2 Evaluation: Effect of Entity Separation

Even though control Entities are separated from management Entities in our proposal, some control and management Entity pairs have to synchronize with each other. For example, in a provider NMS, a “control fiber” is used in an operation to create a route-list for lambda paths. However, at the end of the operation, the provider NMS has to allocate the “management fiber” as a component of a route because a “control fiber” can be a bundle of optical fibers and cannot specify a physical optical fiber. On the other hand, there are control Entities that seldom require synchronization with management Entities. For example, OXC are divided to two Entities namely “management OXC” and “control OXC”, but they are independent from each other, and synchronization is seldom required unless a new optical fiber is assigned to one of the ports of the OXC.

For these reasons, we categorize our proposal into Figs. 7(1) and (2) and compare them with the conventional NMS, which is shown as Fig. 7(3). The case of a control Entity that rarely synchronizes with the corresponding management Entity is represented in Fig. 7(1). The case of a control Entity that has to synchronize at the end of the transaction is represented in Fig. 7(2). The conventional NMS, which has a common Entity for both control and management, is represented in Fig. 7(3). For example, though ITU-T G.805 [17] defines a Link MO, that MO is not divided into two MOs based on control and management functions.

As mentioned above, a control operation tends to take a longer time than a management operation. Thus, we change the operation time of a control operation from 100 to 2000 ms, and the same control operation is repeated. During the repeated control operation, we randomly require a management operation that gets one

attribute of a management Entity and evaluated the management operation processing time. MySQL DB [15] is used and the transaction isolation level is set as “TRANSACTION_SERIALIZABLE” [18] to prevent a discrepancy in the DB.

By and large, management operation processing times under the conditions of Figs. 7(1) and (2) are superior to that under the condition of Fig. 7(3). This is because under the condition of Fig. 7(3), the management operation has to wait once the Entity is locked by the control operation. The management processing time gaps between conditions shown in Figs. 7(1), (2), and the condition shown in 7(3) are not necessarily proportional to the processing times of the control operation. This is because a management operation request is randomly sent; thus, if the request reaches the early part of a control operation, it takes a longer time, but if the request reaches the latter part of a control operation, it takes less time. However, having a common Entity between control and management is clearly not effective for the operation performance. On the contrary, there is almost no gap between Figs. 7(1) and (2). That means that synchronization at the end of the control operation is not burdensome.

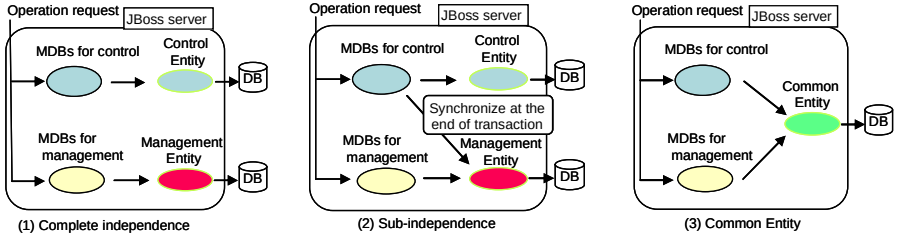


Fig. 7. Evaluation conditions

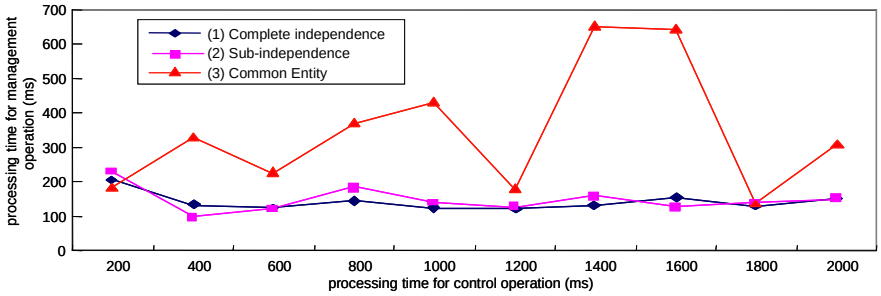


Fig. 8. Evaluation results based on conditions

4 Conclusion

In this paper, we proposed an L1VPN NMS that is customized for each customer. One of the merits of the L1VPN NMS is that the customer operator can manage the VPN from the end-to-end point of view, and even manipulate the lambda path cost depending on the customer policy. When an L1VPN NMS handles fault alarms from a provider network and customer domains, flexible policies are set for dealing with

alarms depending on IP path users. In addition, we demonstrated that separating L1VPN NMSs from a provider NMS significantly reduces the burden of the provider NMS and hastens the entire operation time. We also proposed control Entities that are independent from management Entities. By separating the control Entities from management Entities, controlling processes are free from management processes, and we demonstrated a huge performance improvement in the management process by using our functional architecture compared with the conventional architecture.

References

1. Takeda, T., Inoue, I., Aubin, R., Carugi, M.: Layer 1 virtual private networks: service concepts, architecture requirements, and related advances in standardization. *IEEE Commun. Mag.* 42(6), 132–138 (2004)
2. Bonerjee, A., Drake, J., Lang, J.P., Turner, B.: Generalized multiprotocol label switching: an overview of routing and management enhancements. *IEEE Commun. Mag.* 39(1), 144–150 (2001)
3. Matsuura, H., Murakami, T., Takami, K.: An Interdomain Path Computation Server for GMPLS Networks. *IEICE Trans. Commun.* E88-B(8), 3329–3342 (2005)
4. Oki, E., Shimazaki, D., Shiimoto, K.: Generalized traffic engineering protocol for multi-layer GMPLS networks. *IEICE Trans. Commun.* E88-B(10), 3386–3894 (2005)
5. Ash, J., Roux, J.L.: Path Computation Element (PCE) Communication Protocol Generic Requirements, RFC 4657 (September 2006)
6. Matsuura, H., Takami, K.: Hierarchically Distributed PCE for End-to-End Bandwidth-Assured VPN Service. *IEICE Trans. Commun.* E90-B(11), 3042–3051 (2007)
7. Takeda, T., Ohba, T., Inoue, I., Urushidani, S.: Resource management in layer 1 virtual private networks. *IEICE Trans. Commun.* E88-B(8), 3343–3352 (2005)
8. Takeda, T., Brungard, D., Papadimitriou, D., Ould-Brahim, H.: Layer 1 virtual private networks: driving forces and realization by GMPLS. *IEEE Commun. Mag.* 43(7), 60–67 (2005)
9. Jing, W., Michel, S., Savoie, M., Campbell, S., Bill, A.: Layer 1 virtual private network management by users. *IEEE Commun. Mag.* 44(12), 86–93 (2006)
10. Java Community Process, JSR-000220 Enterprise JavaBeans 3.0 (Final Release), <http://jcp.org/aboutJava/communityprocess/final/jsr220/index.html>
11. Dijkstra, E.W.: A Note on Two Problems in Connexion with Graphs. *Numerische Mathematik* 1, 271–299 (1959)
12. Ma, Q., Steenkiste, P.: On Path Selection for Traffic with Bandwidth Guarantees. In: *Proceedings of IEEE International Conference on Network Protocols* (October 1997)
13. <http://h50146.www5.hp.com/products/servers/proliant/>
14. <http://www.JBoss.org/JBossas/downloads/>
15. <http://dev.mysql.com/downloads/mysql/5.0.html>
16. ITU-T Recommendation for M.3100, Generic network information model (2005)
17. ITU-T Recommendation for G.805, Generic functional architecture of transport networks (2000)
18. <http://wiki.jboss.org/wiki/ConfigDataSources>

Algorithms to Managing Unicast, Multicast and Broadcast Transmission for Optical Switches

Dawid Zydek¹, Leszek Koszalka², Iwona Pozniak-Koszalka², and Keith J. Burnham³

¹ Dept. of Electrical and Computer Engineering, University of Nevada,
Las Vegas, USA

² Chair of Systems and Computer Networks, Wrocław University
of Technology, 50-370 Wrocław, Poland

³ Control Theory and Applications Centre, Coventry University,
CV1 5FB Coventry, United Kingdom
leszek.koszalka@pwr.wroc.pl

Abstract. This paper deals with an optical switching problem concerning an architecture represented by input-output request matrix. New algorithms such as LS, Q-family and J-family are presented and thoroughly studied. J-family algorithms are some modifications of the well known Double algorithm. The study focuses on comparison between the considered algorithms and known algorithms such as Double and Exact. The local comparison criteria consist of the number of decomposed matrices, the number of empty slots, the total time of decomposition and excess. As the global index of performance the weighted penalty is introduced, which comprises local criteria. The properties of the proposed algorithms are shown on examples, where matrices with various elements are decomposed into permutation matrices (switch configurations). An idea of experimentation system and some results of investigations are presented. The properties of the considered algorithms are discussed, especially significant results for LS algorithm are described.

Keywords: Network transmission, algorithm, experimentation system.

1 Introduction

Optical technologies are very popular and still very much under research. They have been developed to increase bandwidth of transmissions and to ensure greater data safety. Today, in the computer networks, optical fibers are the main medium for transmission, but packets are still switched by electronic devices. Optical switching technologies [2], [3], [4] are expensive nowadays, but in the near future these devices will be employed in network centers to manage traffic [8]. Efficient scheduling for optical switches requires proper architecture and special algorithms that optimize final schedule of switching [6]. Architecture dedicated for such devices can operate large number of ports. This architecture might be modelled by the request matrix, in which each row represents an input port while the column represents an output port. Elements of the request matrix are values of time requirements for establishing connection and to transmitting the data between appropriate ports [1], [9]. It must be

mentioned, that at the same time it is not possible to set up connections between all the ports described in the request matrix [2], [3]. To achieve this we have to get information about types of transmission requested by ports and afterwards in unicast traffic case use algorithms that decompose the unicast matrix into permutation matrices, which also represent connection between inputs and outputs. As opposed to the unicast matrix, connections specified by each permutation matrix (called configurations matrices [1]) can exist simultaneously. In the literature (e.g. [1], [8]) many decomposition algorithms are presented and examined, in particular Exact and Double dedicated for optical switches. In this paper, we propose algorithms called J-family and Q-family, which are some modifications of Double and Exact algorithms. Moreover, the new algorithm called Less Switching (LS) is presented.

Authors proposed an experimentation system based on multistage experiment designing [5] for conducting simulation research in order to compare properties of the considered algorithms. Besides well known comparison criteria such as the number of configurations [2] and empty slots [3], we introduce the new criteria such as the decomposition penalty and the excess. The global criteria are the total penalty T_p which constitutes speedup to compensate delays occurred during decomposition process and the weighted penalty W_p which is an intelligent index of performance based on results from previous work [1]. These measures can help switch designers to make a decision, which algorithm will be a better for satisfying a given requirements.

The rest of this paper is organized as follows: The model of architecture is described in Section 2. Section 3 presents criteria for comparison of algorithms. In Section 4 the considered algorithms are described. In Section 5 some results of research are discussed and experimentation system concept is presented. Final remarks appear in Section 6.

2 Model of Architecture

The considered model of architecture [1], [2] operates on the switch with $N \times N$ ports (N inputs and N outputs). Requests of connections between these ports are mapped on the request matrix C . The rows of C are corresponding to the inputs, while the columns to the outputs (Fig. 1).

Elements of C correspond to the requests of switch for connecting appropriate ports. These requests are accumulated over assumed interval T and interpreted as time needed to establish connection and to transmit data between ports. The state, when each input port has set the link with an output port, is called switch configuration.

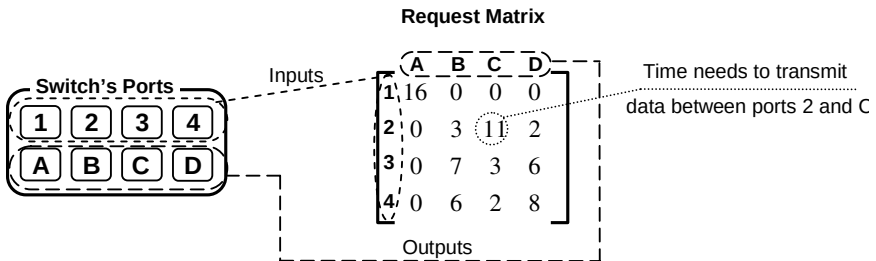


Fig. 1. Modelling connections between ports in the request matrix C

The presented architecture has certain restrictions: (i) no changes of switch configuration while switching elements are transmitting, and (ii) for unicast transmission – in the same time a given input port can establish connection only with one output port. Elements of matrix C_{MB} (*Multicast Broadcast matrix*) also represent time, but the time needed for multicast and broadcast transmission requested by ports. In other words, when data are being sent from one input to group of output ports (i.e. in the same time single input is connected with certain group of outputs), considered non-zero entries mark that type of transmission. The non-zero entries in the row must be equal (the same data are being sent – the same time for transmission). Thus, in the presented system C_{MB} is used only for marking the types of transmission (Fig.2): (i) zeroes entries mean unicast traffic, (ii) non-zeroes entries in whole row mean broadcast traffic, (iii) non-zeroes entries in the part of row mean multicast traffic.

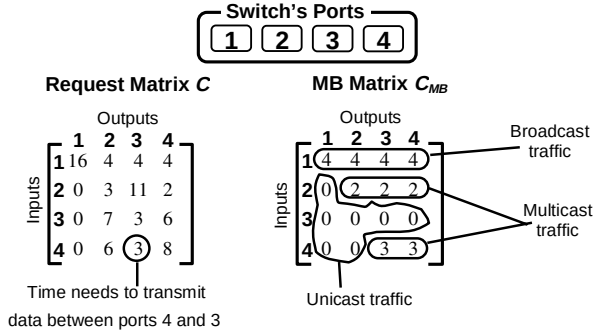


Fig. 2. Example of switch requests using C and C_{MB}

In most cases, when we take into consideration restrictions for unicast transmission, it is not possible to activate transmission basing on the *Unicast Matrix* $C_U = C - C_{MB}$ (see an another example in Fig. 3).

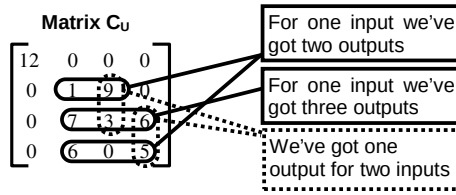


Fig. 3. Example of C_U , where transmission is impossible

Then, before transmission, we have to decompose matrix C_U along with the formula (1), where P - permutation matrix, K - configuration matrix, Φ – weight.

$$C_U = \Phi(1)*P(1) + \Phi(2)*P(2) + \dots + \Phi(i)*P(i) = K(1) + K(2) + \dots + K(i). \quad (1)$$

After decomposition the matrix C_U is replaced by a sum of matrices for multicast and broadcast transmissions as well as the configuration matrices. In practice, the chosen, applied decomposition algorithm affects the efficiency of switch. Evaluation of efficiency can be made using various criteria that are present in the next section.

3 Criteria for Algorithm Quality Evaluation

Number of Configurations N_C . It is defined as the total number of the configuration matrices – see an example in Fig.4.

$$\begin{array}{cccc}
 C_U & = & K(1) & + & K(2) & + & K(3) \\
 \begin{bmatrix} 16 & 0 & 0 & 0 \\ 0 & 3 & 11 & 2 \\ 0 & 7 & 3 & 6 \\ 0 & 6 & 2 & 8 \end{bmatrix} & = & \begin{bmatrix} 16 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 8 \end{bmatrix} & + & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 11 & 0 \\ 0 & 0 & 0 & 6 \\ 0 & 6 & 0 & 0 \end{bmatrix} & + & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 \\ 0 & 7 & 0 & 0 \\ 0 & 0 & 2 & 0 \end{bmatrix}
 \end{array}$$

Fig. 4. Example of decomposition, where $N_C = 3$

Number of Empty Slots N_{ES} . It is defined as a sum (2) of empty slots N_{es} for every decomposed matrix $K(i)$ – see an example in Fig. 5.

$$N_{ES} = N_{es}(1) + N_{es}(2) + \dots + N_{es}(i), \quad \text{where } N_{es} = \max(K(i)) - \min(K(i)) \quad (2)$$

max(K(i)), min(K(i)) – maximum and minimum element of matrix K(i), respectively.

$$\begin{array}{cccc}
 18 & = & 0 & + & 11 & + & 7 \\
 N_{ES} & = & N_{es}(1) & + & N_{es}(2) & + & N_{es}(3) \\
 \begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 3 & 11 & 2 \\ 0 & 7 & 3 & 6 \\ 0 & 6 & 2 & 3 \end{bmatrix} & = & \begin{bmatrix} 3 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \\ 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 3 \end{bmatrix} & + & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 11 & 0 \\ 0 & 0 & 0 & 6 \\ 0 & 6 & 0 & 0 \end{bmatrix} & + & \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 \\ 0 & 7 & 0 & 0 \\ 0 & 0 & 2 & 0 \end{bmatrix}
 \end{array}$$

Fig. 5. Calculation of N_{ES} (here equal to 18), where $N_C=3$, $N_{es}(1)=0$, $N_{es}(2)=11$, $N_{es}(3)=7$

The Excess. The decomposition algorithms used to decompose unicast matrix C_U are completely, when the unicast matrix C_U is less or equal to the sum of all matrices $K(i)$. In the case, when we have got equality, then configurations $K(i)$ exactly cover matrix C_U . However, when sum of $K(i)$ is greater than C_U , our algorithms are also completely, but we are obtaining excess E_{Cu} (3), i.e. it occurs empty connections between some ports (it may cause transmitting empty information).

$$E_{Cu} = \max(C_U - \sum_{n=1}^i K(n)). \quad (3)$$

Decomposition Penalty. For the considered algorithms, the time required to decompose the unicast matrix C_U may be different. Therefore, we introduce the criterion called the decomposition penalty D_p . In the most real experimentation systems the simulations are governed by multi-task operation system. Thus, the value of D_p is expressed in processor units (cycles of processor).

The weighted penalty. The global index of performance is the weighted penalty W_p expressed by (4). It is an extension of T_p presented in [1]. T_p may be regarded as the total speedup required to compensate all delays. W_p can be easily interpreted as a complex measure of algorithms performance.

$$W_p = \varphi_1 N_C + \varphi_2 N_{ES} + \varphi_3 E_{Cu} + \varphi_4 D_p. \quad (4)$$

The proposed values of coefficients in (4) are: $\varphi_1=0.2$, $\varphi_2=0.66$, $\varphi_3=0.01$, $\varphi_4=0.09$. They have been determined by examination of influences of local criteria on the global index T_p . If all weights are equal to 1 then W_p is equal to T_p .

4 Algorithms

The main objective of the presented algorithms is decomposition of the unicast matrix into permutation matrices. It is possible to find in the literature [1], [2], [3] algorithms such as Exact and Double that can solve this problem. Here, we are presenting improvements of these algorithms, as well as our own LS algorithm.

Quick Exact algorithm (QExact). It was created by modification of Exact (Tab. 1).

Table 1. The characteristics of Exact and QExact algorithms

<i>Exact</i>	<i>QExact</i>
▪ set $i=1$ While $C_U \leq 0$ do ▪ $P(i)=0$ repeat 1. find maximum element of matrix C_U, mark its <i>Max</i> if it creating $P(i)$ is possible - save value of <i>Max</i> - create one row of $P(i)$, where no zero element have same position like position of <i>Max</i> in C_U else - go to 1. passing over found elements until $P(i)$ will be created ▪ from all values of <i>Max</i> find minimum element, mark its $\Phi(i)$ ▪ set $C_U = C_U - \Phi(i)*P(i)$, $i = i + 1$	▪ set $i=1$ While $C_U \leq 0$ do ▪ $P(i)=0$, $r = 1$ repeat 1. find maximum element of row no. r in the matrix C_U, mark its <i>Max</i> if it creating $P(i)$ is possible - save value of <i>Max</i> - create row no. r of $P(i)$, where no zero element have same position like position of <i>Max</i> in C_U else ▪ go to 1. passing over found elements ▪ set $r = r + 1$ until $P(i)$ will be created ▪ from all values of <i>Max</i> find minimum element, mark its $\Phi(i)$ ▪ set $C_U = C_U - \Phi(i)*P(i)$, $i = i + 1$

The idea of QExact consists in providing a new way for searching maximum element in the unicast matrix – by searching a single row of the matrix instead of considering all rows. An efficiency of these algorithms is shown in Fig. 6. It may be observed, that decomposition penalty D_p for the QExact algorithm is lower than for Exact. Moreover, it may be noticed, that D_p has an impact on number of configurations N_C .

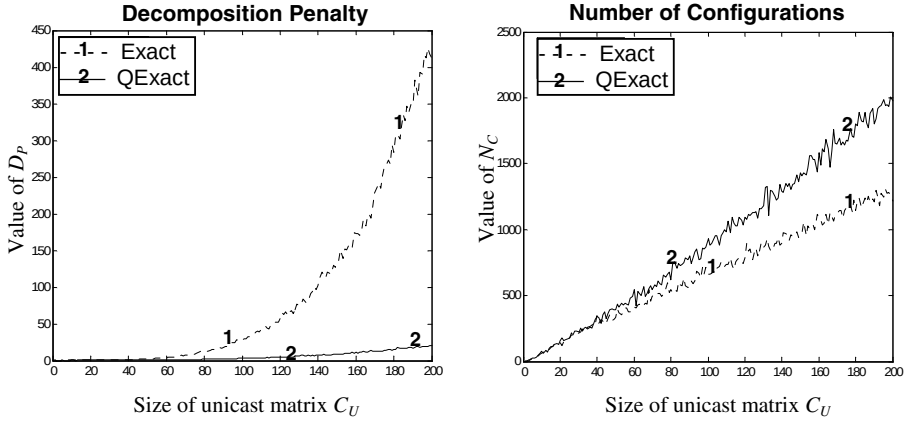


Fig. 6. Decomposition penalty and number of configurations for Exact and Qexact

Less Switching. The algorithms called Less Switching (LS) and Quick Less Switching (QLS) are modified versions of Exact and QExact algorithms, respectively. This modification consists in omitting part of basic algorithm (eliminating next searching procedure), while weight Φ_i in (1) is being found. It causes that D_p is decreased. However, decreasing D_p is a reason of increasing the number of empty slots N_{ES} . The local criteria comparison of these algorithms is shown in Fig. 7.

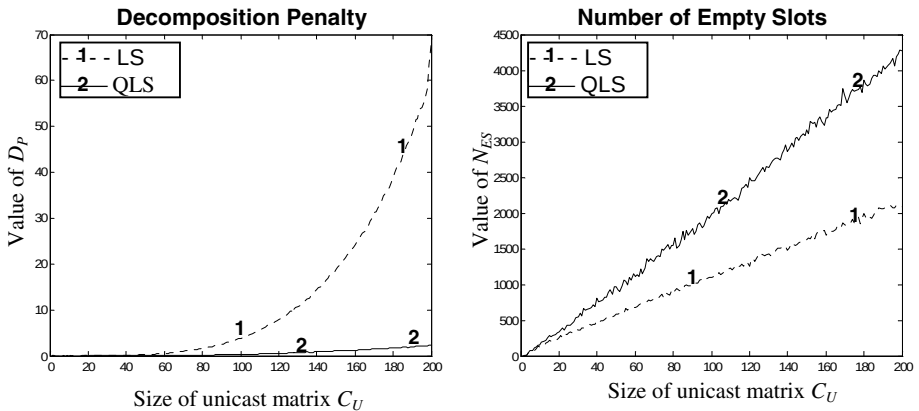


Fig. 7. Decomposition penalty and number of empty slots for LS and QLS algorithms

J-family algorithms. The set of J-family algorithms contains: JExact, JQExact, JLS and JQLS. These algorithms are created on the basis of well known in the literature Double algorithm [1], [2]. Structure of Double consists of two parts: preliminary schedule and final schedule. Weights Φ in (1) are constant. They depend on interval T and size N of the unicast matrix. Double algorithm has non-zeroes excess E_{Cu} . The reasons for the excess are constant weights and non-optimal final schedule. Therefore, during creating J-family algorithms, authors have concentrated on an improvement of the final schedule. It was done by applying (in the second part of Double, when the final schedule is being determined) an another algorithm (see Tab. 2 for details).

Table 2. J-family algorithms

	<i>Double</i>	<i>JExact</i>	<i>JQExact</i>	<i>JLS</i>	<i>JQLS</i>
<i>preliminary schedule</i>	Double	Double	Double	Double	Double
<i>final schedule</i>	Double	Exact	QExact	LS	QLS

That way the flexible weights to the final schedule have been introduced. It caused, that the excess have been eliminated. Using of the proposed ideas provides greater D_p , but other parameters (local criteria) reach better results (e.g. Fig. 8).

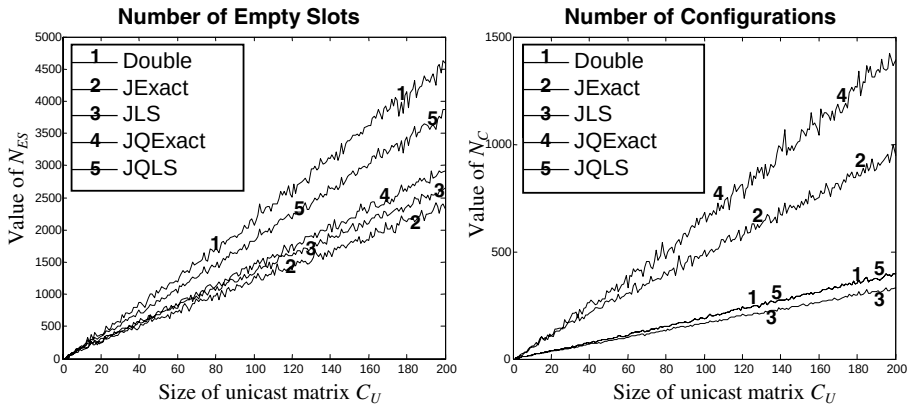


Fig. 8. The local criteria for Double and for J-family algorithms

The process of unicast matrices decomposition made with the considered algorithms and evaluated using the introduced criteria may be treated as input-output system.

5 Investigations

Example of research. During experiments we focused on the comparison of the presented algorithms. Our experimentation system has been implemented in MatLab environment, version 7.0. To investigate properties of algorithms we have used the unicast matrices C_U with range of entries between: $0 \div 5$, $0 \div 10$, $0 \div 15$, and $0 \div 20$. The same configuration of C_U was used for every considered algorithm. The final results of this complex experiment are presented in Fig. 9 (a-d), where 1-Double, 2-Exact, 3-LS, 4-QExact, 5-QLS, 6-JExact, 7-JLS, 8-JQExact, 9-JQLS.

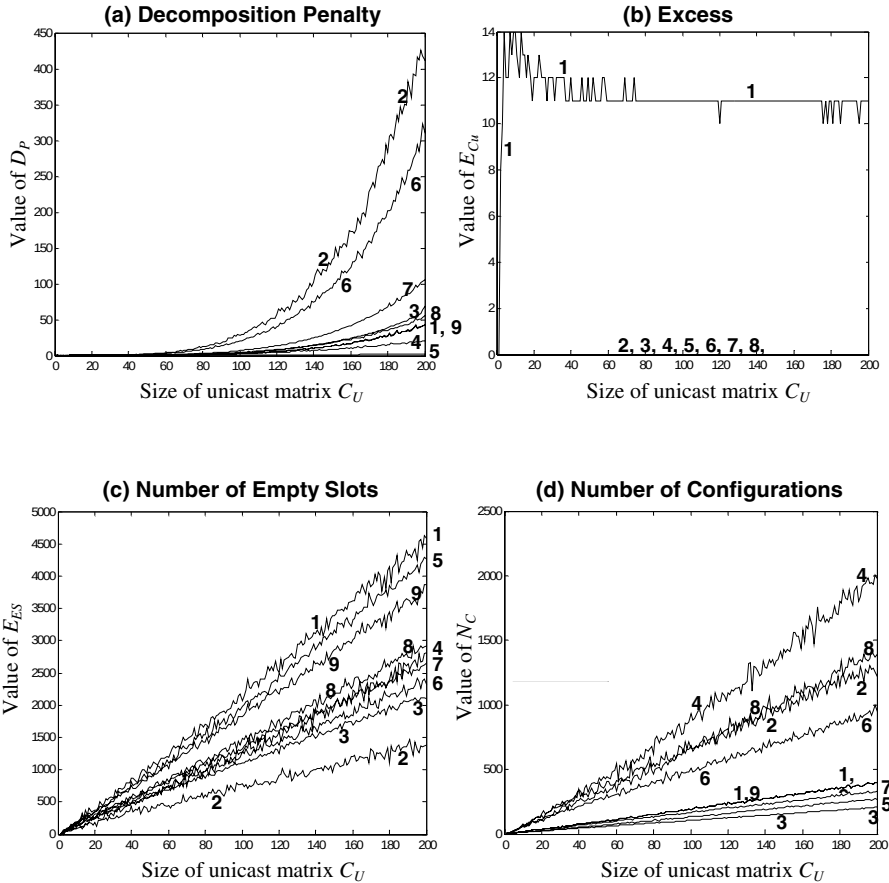


Fig. 9. Results of simulation for the algorithms – local case

It may be observed from Fig. 9a that the smallest D_P was obtained for QLS algorithm. Moreover, it should be noted that for smaller unicast matrices C_U (up to the size of 90×90) all investigated algorithms gave similar results. In opposite, for bigger matrices the differences between results produced by algorithms were remarkable. Fig. 9b confirms that we have got excess only for the Double algorithm. For the rest

of algorithms the excess doesn't exist. In Fig. 9c the number of empty slots is shown. This is important to notice that the values of this parameter (criterion) were greater than for the rest of the considered parameters. The number of configurations (Fig. 9d) also provided greater penalties, but not such significant like number of empty slots.

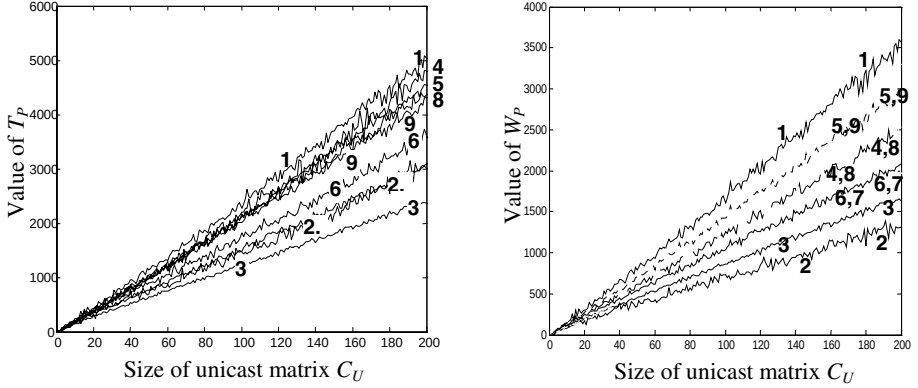


Fig. 10. Results of simulation for the algorithms – global case

Taking into account the global criterion (Fig. 10) we have observed that the smallest the best results were obtained for the LS (see T_P) and Exact (see W_P) and the worst results for Double. The J-family algorithms were placed in the middle.

Future investigations. The evaluation of the considered algorithms required further investigations i.e. implementation of the experimentation system following ideas in [5], [7] with logical structure as input-output system presented in Fig. 11. The system may be described by the relation: $E = R(A, P)$

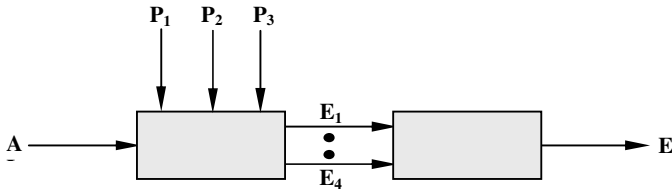


Fig. 11. Block-diagram of the defined input-output system

The elements of the system are:

Controlled input - A : decomposition algorithm being an element of the set $\{\text{Double, Exact, LS, QExact, QLS, JExact, JLS, JQExact, JQLS}\}$;

Problem parameters - P_1 : denoted by N – the size of matrices C and C_{MB} , P_2 : the range of uniformly distributed pseudorandom numbers for matrix C , P_3 : the range of uniformly distributed pseudorandom numbers for matrix C_{MB} .

Local outputs - E_1 : denoted by N_C – the number of configurations, E_2 : denoted by N_{ES} – the number of empty slots, E_3 : denoted by E_{Cu} – the excess, E_4 : denoted by D_P – the decomposition penalty;

Global output E : denoted by W_p – the weighted total penalty.

The designed experimentation system is composed of modules such as *Experiment Designing*, *Calculating Criteria* and *Presenting Complex Results* giving opportunities for carrying out series of experiments in automatic manners.

6 Final Remarks

The proposed algorithms may be utilized in finding the efficient configuration of the computer network, in particular for optical switching with architecture represented by input-output request matrix.

On the basis of experiments taken it has been shown that when the global measure of efficiency is taken then the best results were obtained for LS and Exact algorithms. Moreover, the results of the presented simulation research allow coming to the conclusion that J-family algorithms are promising and they may provide good solution to non-specialized systems.

In our opinion, the *index of performance* expressed by (4) may be regarded as an universal tool in choosing the proper algorithm. The rest of the presented criteria (local criteria) can be also utilized but in case of some specific cases. For instance, the knowledge about algorithm which is coming only from the D_p criterion has not provided very productive information, because influence of this parameter on the total penalty.

The further research in the considered area will be continued with the emphasis on looking for algorithms which should yield decreasing the number of empty slots N_{ES} and decreasing the number of configurations N_C , either.

References

1. Pozniak-Koszalka, I., Zydek, D.: Architecture and Algorithms of Unicast, Multicast and Broadcast Transmission for Optical Switching. In: Proc. of ICN 2007 Sainte-Luce, France, p. 95. IEEE Computer Society Press, Los Alamitos (2007)
2. Towles, B., Dally, W.J.: Guaranteed Scheduling for Switches with Configuration Overhead. IEEE/ACM Trans. on Net 11, 835–847 (2003)
3. Inukai, T.: An Efficient SS/TDMA Time Slot Assignment Algorithm. IEEE Trans. Commun. 27, 1449–1455 (1979)
4. Neukermans, A., Ramaswami, R.R.: MEMS Technology for Optical Networking Applications. IEEE Commun. Mag. 39, 62–69 (2001)
5. Koszalka, L., Lisowski, D., Pozniak-Koszalka, I.: Comparison of Allocation Algorithms for Mesh - Structured Networks with Using Multistage Simulation. In: Gavrilova, M.L., Gervasi, O., Kumar, V., Tan, C.J.K., Taniar, D., Laganá, A., Mun, Y., Choo, H. (eds.) ICCSA 2006. LNCS, vol. 3984, pp. 58–67. Springer, Heidelberg (2006)
6. Crescenzi, P., Deng, X., Papadimitriou, C.: On Approximating a Scheduling Problem. Journal of Combinatorial Optimization 5, 287–297 (2001)
7. Koszalka, L.: Static and Dynamic Allocation Algorithms in Mesh Structured Networks. In: Madria, S.K., Claypool, K.T., Kannan, R., Uppuluri, P., Gore, M.M. (eds.) ICDCIT 2006. LNCS, vol. 4317, pp. 39–50. Springer, Heidelberg (2006)
8. International Engineering Consortium, <http://www.iec.org>
9. Kasprzak, A.: Packet Switching Wide Area Networks, WPWR, Wrocław /in Polish/ (1999)

Network Planning and Adaptive Routing for Multimedia Traffic

Priscila Solís Barreto¹, Paulo H. P. de Carvalho², Rafael Dias Oliveira³,
and Maximiliano Prestes Ceppo³

¹ Department of Electrical Engineering

² Department of Computer Science

University of Brasilia, Brasília-DF, Brasil

pris@cic.unb.br, paulo@ene.unb.br, {rafael,max}@labcom.unb.br

Abstract. This work presents a methodology for network traffic engineering based on a hybrid traffic model and an adaptive routing algorithm. The hybrid traffic model pursues the calculation of more accurate QoS metrics and effective bandwidth for multimedia traffic in order to optimize the sizing of network elements at a minimal cost. During network operation, the links weighted values are modified to compute shortest path routing algorithms in an adaptive manner to comply with the traffic effective bandwidth and QoS requirements. The network routing methodology was evaluated in a network topology that aggregates different traffic types, Poisson and self-similar. The results show that the proposed methodology achieves the QoS requirements and promotes a more efficient traffic balancing within the network.

Keywords: Traffic characterization, adaptive routing, QoS metrics, optimization.

1 Introduction

The multimedia traffic results when integrating, over a single communication channel, a diverse range of traffic sources [1,2]. Different observations of operating networks confirm that the aggregated traffic in the cores, i.e. the multimedia traffic, has a self-similar and multifractal nature [3,4,5,6,7,8,9] which results from multiplexing different sources, that significantly differ in their traffic finite buffer systems when compared to the Markovian traffic sources.

Also, observations show that the existence of noticeable concentrated periods of contention and idleness of self-similar traffic can greatly increase packet delay and loss probability and thus reduce system resource utilization [11]. Also, several research works point that nowadays the research problems and challenges for network traffic may be grouped into three areas: workload characterization, performance analysis and traffic control [10].

Some models have been developed to treat traffic characterization of self-similar and multimedia traffic [11,12,13]. Some of these works appear to have a certain tendency to overestimate the QoS values while others rely on the extraction of more than three traffic parameters to perform the characterization. Clearly, this overestimation has a negative impact when trying to optimize network planning.

Along with the network planning procedures, routing strategies are a fundamental issue to optimize network resources. The traditional approach of routing protocols such as OSPF (Open Shortest Path First) rely in the use of dynamic operations to be adaptable to topology changes and distribute link state information within a certain domain in order to define network paths [25]. The computation of the shortest path between two nodes is based on a metric link. In most of the commercial implementations, this metric does not consider the current network state and the different QoS traffic requirements [24]. Several works indicate that the setting of link metrics i.e. the link weights, when based on traffic measurements and topology data collected from the operational network, may improve traffic balancing and promote network stability [14,15,16]. Many of these works do not consider the traffic fractal properties. We propose that the knowledge of these properties may improve the workload characterization and performance analysis to establish adaptive link weights that will result in better adaptive routing strategies [17].

This article proposes a methodology for the calculus of effective bandwidth to define link weights and to perform an adaptive routing strategy based on the OSPF algorithm. The calculus of effective bandwidth uses a hybrid traffic model for multimedia traffic. This proposal pursues the improvement of workload network characterization and performance analysis to achieve a better traffic network balancing and consequently, a better network resource utilization.

The article is organized in the following manner: section 2 presents a discussion about the fundamentals of multimedia traffic engineering and describes the theoretical basis of the hybrid traffic model used for the calculus of effective bandwidth. In section 3, the proposed methodology is evaluated with a simulated network scenario. Section 4 presents the conclusions and future work of this research.

2 Multimedia Network Traffic Engineering

Traffic engineering may be defined as the mechanisms that are used to re-optimize network performance on traffic variation time scales. Several works focus on traffic balancing and characterization to achieve traffic engineering objectives [16, 17].

In Figure 1, a diagram of a proposed traffic engineering framework is described. The Network Planning task is the first step of the proposed TE framework. This task relies in the problem formulation for capacity assignment in a network as shown in [18], which considers a fixed routing on a given network topology. The Network Planning task as shown in Figure 1, characterizes the traffic flows and evaluates the topology to comply with the QoS flows requirements. This task uses a Performance Evaluation procedure to optimize the sizing of network elements, i. e. buffers and links capacities. After that procedure, the framework initiates the second step, the Network Operation task. When new flows arrive to the network, their QoS requirements are evaluated through the Admission Control procedure that should find the better path within the network for the new flow. A procedure for optimizing the path assignment to these new applications should use the calculus of QoS metrics as a criterion, in such a way that the network operation is guaranteed for certain QoS limits, which results in network operation stability.

As can be seen in Figure 1, the Control Admission procedure relies as well on the Performance Evaluation procedure. If the QoS requirements for that flow can not be achieved, the Routing Optimization procedure is triggered and the network routing model is updated. The variables to implement the methodology are summarized in Table 1.

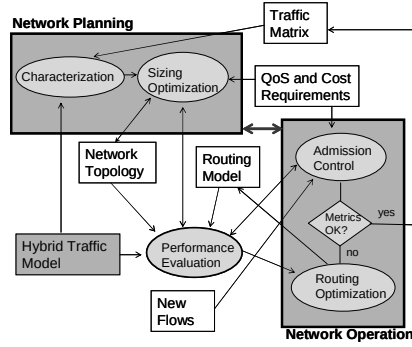


Fig. 1. Traffic Engineering Framework

Table 1. Variables for Traffic engineering with traffic characterization

Given	A network topology A traffic matrix R_H The cost function for link capacities $d_i(C_i)$ A set of traffic trunks defined by $FECs=F$
Minimize	A cost function
With the variables	$VP_R=(L_1, L_2, \dots, L_b)$ $C=(C_1, C_2, \dots, C_n)$ $F_L=\{(f_i, l) \in M:F \rightarrow VP_R\}$
Restrictions	$\sum_i (f_i, L) < \rho(L)$ $\forall f_i \in (f_i, L), QoSM(f_i, L) \leq QoSR(f_i)$, $L \in VP_R$ and exists a $f_l \in F$ where $f_i=f_l$

In Table 1, the $QoSR$ function defines the QoS requirements of a specific service class and the $QoSM$ function defines the QoS values of that traffic class when mapped in a VP (virtual path). The QoS parameters are the delay and packet loss probability. The $\rho(L)$ function defines the maximum admissible load for the link L . The set of trunks F are the traffic flows grouped by FEC (Forwarding Equivalence Classes) which are mapped in virtual paths when using the function M .

Each flow f_n is characterized by a group (S, A, P) in which $S=(H, a, \sigma^2)$, with H as the Hurst parameter for the traffic trace, a is the mean rate and σ^2 is the variance. The parameters A and P are the maximum delay and the maximum loss admissible for the flow, calculated with function $QoSR$.

The above definitions induce that, for a flow to be associated to a FEC, it should comply the following conditions:

- (i) $\forall f_i \in T_m \in FEC_s, Ag_H(H_i, T_m) < H_{FEC}$
- (ii) $QoS\_Delay(f_i, T_m) < A_{FEC}, QoS\_Loss(f_i, T_m) < P_{FEC}$

In item(i) the function Ag_H estimates the aggregated H when joining the f_i flow into the T_m trunk. The resulting value of Ag_H should be lower then the expected value for that FEC, represented by the value H_{FEC} which is the maximum H associated to the class. The variable A_{FEC} is the maximum delay and P_{FEC} is the maximum packet loss probability. The function QoS_Delay calculates the delay for the T_m traffic trunk when the f_i flow is mapped within this path, and the QoS_Loss function calculates the packet losses.

For a better understanding, a description of the Control Admission procedure is shown in the following 6 steps:

```

Control Admission( $f_i(H, A, P), G(V, E, L), T_D$ )
1 ClassType  $\leftarrow$  ClassFlow( $f_i(H, A, P)$ )
2  $B \leftarrow$  Mapflow( $f_i(H, A, P), G(V, E, L), ClassType$ )
3 If  $B = 1$ 
4   then UpdateTable( $T_D, f_i$ )
5       AcceptFlow( $f_i, G(V, E, L)$ )
6   else RoutingOptimization( $G(V, E, L), T_D$ )

```

The algorithm receives as parameters the traffic flow, the network topology and a table T_D which contains the definitions of the flows mapping and their associated classes. Line 1 of the algorithm makes an identification of the traffic class that should be associated to the traffic flow, based on its QoS metrics specifications. Line 2 maps the flow in one available virtual path, in which the QoS parameters and the link load are evaluated to verify that accomplishment of the QoS restrictions. Then, this step is an optimization process which should find the best virtual path to map the traffic flow.

The variable B of step 2 is used in step 3 to trigger the routing optimization procedure. If the returned value is 1, the flow mapping was successfully achieved. Even though the flows mapping procedure preserves the current routing paths, its intention is to induce a more lasting period of operation stability prior to the beginning of the routing optimization procedure, due to the impact that this procedure may have in the operational network.

As can be seen in Figure 1 and in the previous algorithm, the traffic characterization is a fundamental procedure for the proposed methodology. This methodology uses a hybrid traffic model, which allows the calculus of more accurate QoS metrics for multimedia traffic.

Previous works [19,20] presented a hybrid traffic model which combines theoretical elements of the fBm (Fractional Brownian Motion) and the Markovian models. The hybrid traffic model is based on the following premises:

- The traffic flows are described by the variables (H, a, v) , where H is the Hurst parameter, $0 < H \leq 0.95$, a is the mean and v is the variance coefficient.
- For a traffic load in the network links lower than 30%, independently of the H value or when the H value is lower than 0.5, the QoS values are calculated with the M/M/1/ k model, in which k denotes the buffer size.
- For link loads between 30% and 100%, the QoS values are calculated with a deterministic fBm envelope divided by the channel capacity c , as proposed in 13.

- d) For packet loss probability, when it occurs in a node, the traffic flow that arrives to the following node in the virtual path suffers an influence of this loss since the H value depends on the value of the aggregated variance. We propose the calculus of the media value by preserving the value of the variance coefficient, which is the relation between the mean value and variance.

Within the traffic engineering methodology, the routing strategy defines an adaptive dynamic based on effective bandwidth estimation. The effective bandwidth [21,22] definition for self-similar flows when characterized as fBm flows [12,13] is shown in (1), with m as the mean, $K(H)=H^H(1-H)^H$, a is the variance coefficient, H is the Hurst parameter and P_{loss} is the loss probability with a buffer size b .

$$EB = m + K(H) \sqrt{-2 \ln P_{loss}}^{1/H} a^{\frac{1}{2H}} b^{\frac{-(1-H)}{H}} m^{\frac{1}{2H}} \quad (1)$$

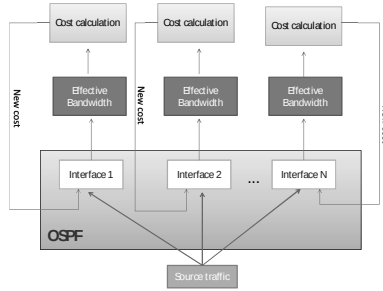


Fig. 2. Adaptive Routing Scheme

In Figure 2 the adaptive routing procedure is described. While the packets flow in the different node interfaces, the number of packets during the time interval s as well as their packet sizes are kept by the effective bandwidth block. In this block, a monitor that tracks the link occupancy returns the link load. Based on the definition of the hybrid traffic model described above the values of effective bandwidth are calculated.

After the effective bandwidth calculus, its value is parameter to the OSPF module, which modifies the link cost to the relation (effective bandwidth/link capacity). The alteration is done simultaneously in all the interfaces. After this alteration, the common dynamics of the OSPF algorithm modify the network paths within the network.

3 Simulation Results

To evaluate the proposed methodology, the network scenario shown in Figure 3 was evaluated. Considering the proposed methodology in Figure 1, the network planning task has already been done for this network scenario. Due to space restrictions in this

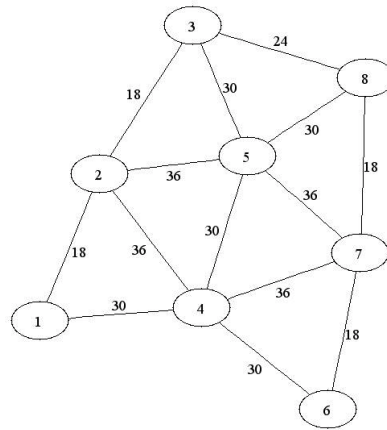


Fig. 3. Experimental Network Topology

Table 2. Traffic flows parameters

Flow ID	Source Node	Destiny Node	First Window	H	Mean Rate (kBps)	Variance (kBps)
1	1	8	1	0.75	12.5	28354
2	2	8	11	0.5	8	16375
3	4	3	1	0.5	10	24989
4	4	8	11	0.85	7.5	19648
5	6	3	11	0.60	7.5	9219
6	7	3	1	0.80	8	48406

article, the procedure and results are not shown in this paper but their accuracy may be consulted in [19] for one QoS metric and the calculus of the link capacity and in [20] for two QoS metrics and the calculus of link capacity and buffer sizes.

In order to verify the results through simulation, the NS-2 (Network Simulator 2) was used. Due to an implementation facility, the simulation was divided in several time windows. The results of a window, i.e. the adapted link costs, are the input to the next window. At the end of the simulation period, the time windows are joined in order to reflect the complete adaptive simulation. The first simulation window receives the links costs defined for the first calculus of the OSPF algorithm. The link costs initially are calculated as the inverse of the link capacity.

In Table 2 are described the traffic traces. The series were generated with traffic generation software developed within this research [23]. The time scale for all is 0.1 seconds and the duration is 400 seconds. All the flows have different parameters, with two of them defined as Poisson ($H=0.5$) and the others as self-similar.

As can be seen in Table 3, the flows 2, 4 and 5 arrive later than the flows 1, 3 and 6. This means that the `ControlAdmission` algorithm described in section 2 was applied until the step 3 in which the returned value for B is 1. So the `RoutingOptimization` procedure should be triggered in order to reposition the flows within

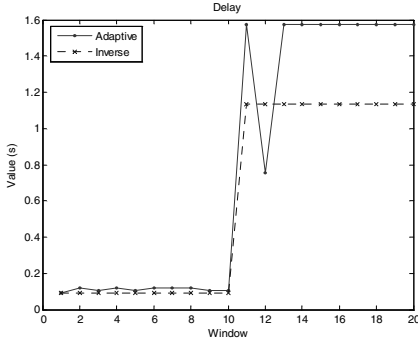


Fig. 4. Delay for flow 1

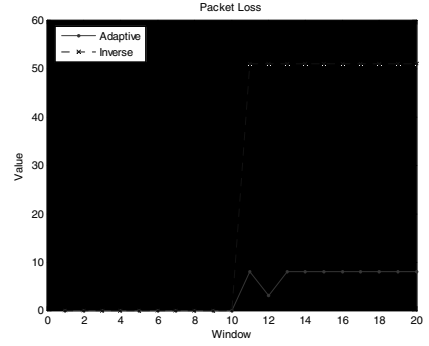


Fig. 5. Packet losses for flow 1

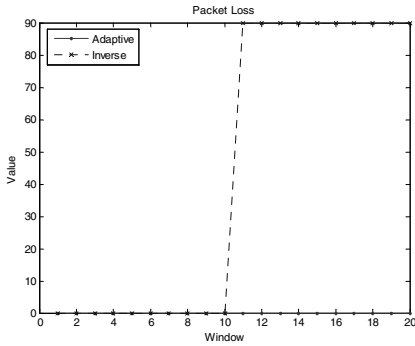


Fig. 6. Packet losses for flow 3

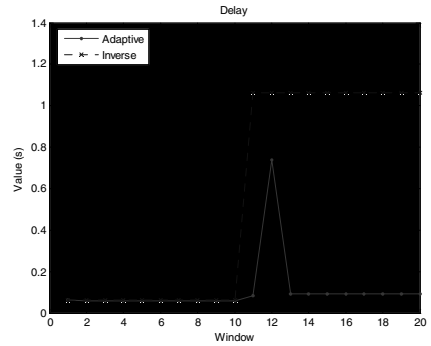


Fig. 7. Delay for flow 3

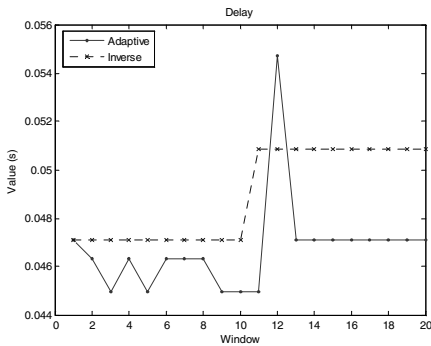


Fig. 8. Delay for flow 6

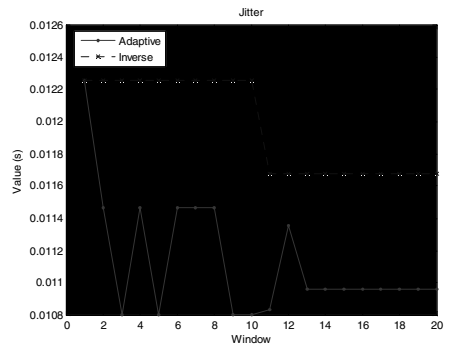


Fig. 9. Jitter for flow 6

the network after modifying the link costs when considering the effective bandwidth necessity. As mentioned before, this procedure relies on the traffic characterization with the hybrid traffic model described in section 2. The total simulation time has 20 windows of 20 seconds each.

The delay values and packet loss for flow 1 are shown in Figure 4 and Figure 5 respectively. For the delay values in the first 10 windows, the difference between the static links cost function (called of inverse in the graph because the cost is commonly defined as the inverse of the link capacity) and the adaptive function based on the effective bandwidth calculation, does not have almost any difference.

The above observation indicates that the setting the routes considering the traditional approach of the OSPF algorithm in the first moment may be correct. After the 10th window, the adaptive procedure results in a higher delay for the flow 1.

The above result is acceptable if the delay relies under the maximum delay admissible for that flow. But when observing the results of packet losses (Figure 5), it appears that the adaptive routing procedure reduced significantly the values when compared to the traditional metric setting. The reason of this result for some flows is due to the fact that for the inverse cost calculus, the routing paths trend to concentrate the flows in the core network links, i.e., the paths that have fewer network hops to the destiny. But, even though the delay may increase for some flows, if it remains under the admissible value, the result is acceptable. For the adaptive routing method, the calculated cost promotes a wider utilization of the border links, which results in a considerable diminution of packet losses.

The results of delay and packet losses for flow 3 are shown in Figure 6 and Figure 7 respectively. For both cases, the adaptive routing procedure reduced all the QoS metrics for the flow. In the case of packet loss the reduction is very significantly. The same observation appears for the flow 6 delay value in Figure 8. For this flow the packet losses were 0 for both cases. The mean jitter for flow 6 appears to have a lower value for the adaptive routing algorithm, as shown in Figure 9.

In Table 3 are shown the comparative results for mean delay, mean jitter and number of packet losses for all the flows, for the adaptive and inverse weighted link cost. Since the jitter has not been used a restriction metric in our model, its results are used just as a reference variable for the other QoS metrics. As can be seen in Table 3, the reduction of packet loss has the most significantly reduction for almost all the cases when applied the adaptive routing algorithm. The results are always lower than the ones for the traditional inverse cost calculus. The mean delay for some flows increase with the adaptive routing algorithm, but as long as it remains under the QoS limit value, the result may be considered acceptable.

Even when the jitter was not a restriction variable considering in the adaptive routing process, it can be observed in Table 3 that for most of the traffic flows, the jitter was also reduced when applying the proposed methodology.

Table 3. Comparative Traffic QoS Results

Flow	Mean Delay (seconds)		Mean Jitter (seconds)		Packet Losses (no. of packets)	
	Inv	Adap	Inv	Adap	Inv	Adap
1	0.5846	0.7625	0.0892	0.0329	510	75
2	0.0489	0.1334	0.0106	0.0121	0	0
3	0.5353	0.1028	0.1078	0.0116	900	0
4	0.9922	1.3094	0.1861	0.0859	740	174
5	1.0291	0.0716	0.0816	0.0104	240	0
6	0.0466	0.0445	0.0114	0.0106	0	0

4 Conclusions

This article proposed a methodology for network planning and adaptive routing for multimedia traffic. The methodology relies on a hybrid traffic model for multimedia traffic. The traffic model characterizes the traffic to produce more accurate QoS metrics that may be used to optimize the cost of network elements in order to reduce the network planning cost. Within the proposed methodology, the routing optimization procedure enters in operation when the control admission procedure detects the impossibility to map a new traffic flow within the established virtual paths. The routing optimization procedure uses an adaptive routing strategy. The adaptive routing strategy modifies the weighted links costs in a dynamic approach based on the effective bandwidth necessity. The calculus of the effective bandwidth relies in the hybrid traffic model and the OSPF algorithm is used to compute the network paths.

The proposed methodology is evaluated in an optimized network with a pre-established traffic matrix and a pre-defined routing strategy. The results show that when the network is in operation, the adaptive updating of weighted links costs when compared to the static approach, results in a higher degree of load balancing which improves the maintenance of the traffic QoS requirements for almost all the traffics. The results were specially improved for the packet loss values. The future work of this research will evaluate the proposed methodology for multifractal traffic.

References

1. Rosen, E., Viswanathan, A., Callon, R.: Request for Comments: 3031. Janeiro (2001)
2. Evans, J., Filsfils, C.: Deploying IP and MPLS QoS for multiservice networks. Morgan Kaufmann, San Francisco (2007)
3. Leland, W.E., Taquu, M.S., Willenger, W., Wilson, D.V.: On the self- similar nature of ethernet traffic (extended version). *IEEE/ACM Trans. on Networking* 2(1), 1–15 (1994)
4. Park, K., Willinger, W.: Self-Similar Network Traffic and Performance Evaluation. John Wiley and sons, Chichester (2000)
5. Sahinoglu, Z., Tekinay, S.: On multimedia networks: self-similar traffic and network performance. *IEEE Communications* 37(1), 48–52 (1999)
6. Yusheng, JI.: Multi-Scale Internet Traffic Analysis Using Piecewise Self-Similar Processes. *IEICE Transactions in Communications* E89-B, 2125–2133 (2006)
7. Crovella, M.E., Bestavros, A.: Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes. *IEEE/ACM Transactions on Networking* 5, 835–846 (1997)
8. Rodrigues, L., e Guardieiro, P.R.: A Spatial and Temporal Analysis of Internet Aggregate Traffic at the Flow Level. In: *Global Telecommunications Conference*, vol. 2, pp. 685–691 (2004)
9. Carvalho, P., Solis, P., et al.: A Traffic Analysis per Application in real IP/MPLS Service Provider Network. In: *2nd IEEE/IFIP International Workshop on Broadband Convergence Networks (BCN 2007)*, pp. 1–5 (2007)
10. Park, K., Willinger, W.: Self-Similar Network Traffic and Performance Evaluation (2000); ISBN: 9780471206446
11. Sheluhin, O., Smolskiy, S., Andrew, O.: Self-Similar Processes in Telecommunications. Wiley Publishers, Chichester (2007)

12. Norros, I.: On the use of fractional Brownian motion in the theory of connectionless networks. *Journal on Selected Areas in Communications* 13, 953–962 (1994)
13. Fonseca, N.L.S., Gilberto, M., e Neto, C.A.V.: On the equivalent bandwidth of self-similar sources. In: *ACM Transactions on Modeling and Computer Simulation*, vol. 10(2), pp. 104–124 (2000)
14. Lee, K., Toguyeni, A., Noce, A., Rahmani, A.: Comparison of Multipath Algorithms for Load Balancing in a MPLS Network. In: Kim, C. (ed.) *ICOIN 2005. LNCS*, vol. 3391, pp. 463–470. Springer, Heidelberg (2005)
15. Elwalid, A., Jin, C., Low, S., Widjaja, I.: MATE: MPLS Adaptive Traffic Engineering. In: *Infocom 2001*, pp. 1300–1309 (2001)
16. Fortz, B., Rexford, J., Thorup, M.: Traffic engineering with traditional IP routing protocols. *IEEE Communications Magazine* 40(10), 118–124 (2002)
17. Fortz, B., Thorup, M.: Robust optimization of OSPF/IS-IS weights. In: *Proc. INOC 2003*, pp. 225–230 (2003)
18. Gerla, M., e Kleinrock, L.: On the Topological Design of Distributed Computer Networks. *IEEE Transactions on Communications* 25(1), 48–60 (1977)
19. Solis Barreto, P., de Carvalho, P.H.P., Soares, A.M., Abdalla Jr., H.: A Traffic Model for Multimedia Applications in Converged Networks. In: *13th IEEE Modelling, Analysis and Simulation in Computer and Telecommunications*, pp. 163–170. IEEE Computer Society, Los Alamitos (2005)
20. Solis Barreto, P., de Carvalho, P.H.P.: Hybrid Traffic Model for Multimedia Network Performance Evaluation. In: *AINA - 22nd International Conference on Advanced Information Networking and Applications-Workshops*, Okinawa, Japão. IEEE Computer Society, Los Alamitos (2008)
21. Kelly, F.P.: Notes on effective bandwidth. In: Kelly, F.P., Zachary, S., Ziedins, I.B. (eds.) *Stochastic Networks: Theory and Applications*, pp. 141–168. Oxford University Press, Oxford (1996)
22. Kesidis, G., Walrand, J., Chang, S.: Effective Bandwidths for Multiclass Markov Fluids and Other ATM Sources. *IEEE/ACM Transactions on Networking* 1(4), 424–428 (1993)
23. Solis Barreto, P., de Carvalho, et al.: Open Source Software for Evaluation of Applications in an Experimental Testbed for Converged Networks. In: *TRIDENTCOM 2006*, p. 6 (2006)
24. Khanna, A., Zinky, J.: The revised ARPANET routing metric. *ACM SIGCOMM Computer Communication Review* 19(4), 45–56 (1989)
25. Moy, J.: OSPF Version 2, RFC 2328 (accessed 14th, July 2007), <http://www.faqs.org/rfcs/rfc2328.html>

Network-Wide Rollback Scheme for Fast Recovery from Operator Errors Toward Dependable Network

Daisuke Arai, Kiyohito Yoshihara, and Akira Idoue

KDDI R&D Laboratories Inc., 2-1-15 Ohara Fujimino City
Saitama Prefecture 356-8502, Japan
{di-arai,yosshy,idoue}@kddilabs.jp

Abstract. Network failures may have a major impact on our society. There are many possible causes of network failures, of which the most significant is operator errors. Consequently, the development of new network management schemes to tackle operator errors is important. We have already proposed a basic idea of a new network-wide rollback scheme to tackle operator errors. In the proposed scheme, we introduce a server to manage historical versions of sets of device configuration. An operator rolls back a set of device configuration via the server when the operator detects a network failure. In this paper, we present a detail of the network-wide rollback scheme. In addition, we provide three rollback procedures, and implement a prototype system to evaluate their rollback time. The proposed scheme will serve for fast recovery from operator errors, as the minimum rollback time is about 41 seconds, when 50 routers are rolled back.

Keywords: Configuration Management, Operator Error, Dependable Network.

1 Introduction

Networks including the Internet are becoming part of our social infrastructure, such as the electricity and water supply. Currently, networks are used not only privately but also for mission critical tasks within companies. With this in mind, network failures may have major impacts on our society and economy.

There are many possible causes of network failures, of which the most significant is operator errors [1][2]. Furthermore, the majority of the operator errors, which we focus on throughout the paper, stems from configuration tasks [2]. Consequently, the development of new network management schemes to tackle operator errors caused by configuration is important to make network more dependable.

In this paper, we consider the dependability as a collective term for availability and its influencing factors [3]. This implies that we need to increase Mean Time Between Failure (MTBF) or to reduce Mean Time To Repair (MTTR), in order to realize the dependable network, since availability is defined by the formula below:

$$\text{Availability} = \text{MTBF} / (\text{MTBF} + \text{MTTR}).$$

Research and development work into the dependable network have been conducted. We can divide them into two main categories: one is by increasing MTBF and the other is by reducing MTTR.

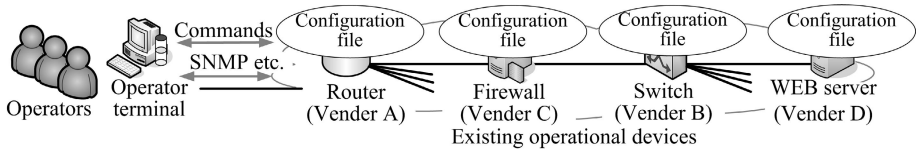


Fig. 1. Typical multivendor network environment

A class of redundancy schemes [4] to increase MTBF is well-known. These schemes supply backup entities to fail-over. When a network failure occurs, these schemes detect the failure, and change over to the backup entities. However, it is difficult for them to address a network failure caused by operator errors, because it is generally hard for devices to proactively detect operator errors. An example is an inadvertent input “100” instead of the correct one “10”, whereas both are syntactically correct for the device operation.

The Pinpoint [5][6] is one of the fast failure detection schemes developed to reduce MTTR. It focuses on component-based Internet services, which are a collection of components developed in the Java 2 Enterprise Edition (J2EE) framework. A user request is processed by certain of such components. The pinpoint adds monitoring software to all the components. It traces each request, together with its result (success and failure), and decides which components are used for the processing. The pinpoint identifies faulty components via standard data mining techniques on the trace. However, this scheme requires all devices to adopt additional monitoring software, which represents an obstacle to deploy this scheme within a practical network, because adding such software to existing operational devices is often difficult.

The undoable email [7] is one of the fast recovery schemes developed to reduce MTTR. It provides an undo layer and non-overwriting storage. The undo layer is provided in the form of a proxy layer that wraps an existing, unmodified mail server. During normal operation, the undo layer snoops mail traffic and stores it to the non-overwriting storage. Meanwhile, an operator restores the mail server to its previous state using the information concerning the non-overwriting storage when the operator inadvertently deletes a user mail. However, this scheme requires a system that is supposed by the schemes, to implement the undo layer for each OS version and protocol, which represents an obstacle to deploy this scheme within a practical network, because in the practical network, a multivendor network environment is common for economic reasons, and it takes considerable time to develop the undo layer for every OS version and protocol.

We have already proposed a basic idea of a new network-wide rollback scheme for fast recovery from operator errors [8]. In this paper, we present a detail of the network-wide rollback scheme. In addition, we provide three procedures to realize the proposed scheme, and implement a prototype system to evaluate three procedures.

This paper is organized as follows: In Section 2, we present an overview of the typical network on which we focus on throughout the paper. We provide the basic idea for the network-wide rollback scheme and address the technical issues involved in Section 3. In Section 4, we present a detail of the network-wide rollback scheme toward the dependable network, while in Section 5, we implement a prototype system. Finally, in Section 6, we evaluate the proposed scheme in terms of MTTR.

2 Typical Multivendor Network Environment

We focus on a multivendor network environment, including existing operational devices, as shown Fig.1. One or more operators manage the network, which consists of devices such as routers produced by vendor A, switches by vendor B, a firewall by vendor C and a WEB server by vendor D. These devices could be generally deployed before the introduction of a network management scheme. An operator enters one or more commands into one or more devices through an operator terminal when performing management tasks. Devices have one or more configuration files and the command entered may change some or all of the configuration files immediately.

An operator monitors whether devices are working normally via management protocols such as the Simple Network Management Protocol (SNMP). When the operator detects a network failure by monitoring results, the operator tries to identify the cause of the network failure, and enters commands into devices in order to recover from the network failure.

3 Basic Idea and Technical Issues

3.1 Basic Idea of New Network-Wide Rollback Scheme

The basic idea of a new network-wide rollback scheme comes from a straightforward fact: for any operational device, its observable behavior is identical whenever the same configuration file is applied. With this in mind, we can expect fast recovery from network failures caused by operator errors, by network-wide rollback, i.e. rolling back the current configuration files of devices concerned to previous ones. We differentiate the network-wide rollback from existing schemes in Fig. 2.

In existing schemes, the time required for recovery from a network failure adds up to a total of: (1) the time for the failure detection (Fig.2 (T1)), (2) the time for the cause analysis (Fig.2 (T2)) and (3) the time for the recovery operation (Fig.2 (T3)).

In the network-wide rollback, the time required for recovery from a network failure can be estimated as the sum of: (1) the time for the failure detection (Fig.2 (T4)) and (2) the time for the network-wide rollback (Fig.2 (T5)). Following the recovery, the operator analyses the cause of network failure (Fig.2 (T6)). If the time of the network-wide rollback (Fig.2 (T5)) is shorter than the sum of the time of the cause analysis (Fig.2 (T2)) and the time of the recovery operation (Fig.2 (T3)), the network-wide rollback will be able to realize the dependable network by reducing MTTR.

3.2 Technical Issues

The following technical issues should be addressed to achieve the network-wide rollback, by studying the typical network in Section 2:

1. The network-wide rollback should perform in a multivendor network environment (Issue 1).
2. The network-wide rollback should be deployed with no modification to existing operational devices (Issue 2).

3. The network-wide rollback should consider the rollback sequence, since the startup-sequences of certain devices depend on others (Issue 3).
4. The network-wide rollback should consider the addition and deletion of devices, since hardware construction of a network is changed (Issue 4).

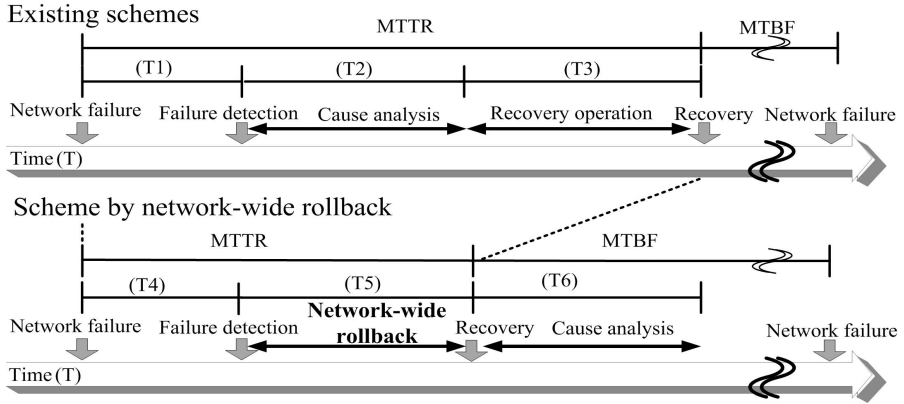


Fig. 2. Differentiation of network-wide rollback from existing schemes

4 Details on Network-Wide Rollback Scheme

4.1 Preparations and Design Principles

Preparations

We make the following preparations.

1. An operator installs a new management server to achieve the network-wide rollback. The server hooks all command requests to and responses from operational devices.
2. For each device, an operator provides the following information:
 - (1) A device name: a character string given by an operator,
 - (2) An IP address or a Fully Qualified Domain Name (FQDN) of a device,
 - (3) A pair of a login ID and its associated password for a device,
 - (4) Get procedure (See below),
 - (5) Rollback procedure (See below),
 - (6) A rollback priority (See below).

An operator should register one or more pairs of get and rollback procedures for each configuration file of a device, with the management server. Each procedure allows us to get a configuration file of the device and to rollback a configuration file to a past one. We will show the detail of these procedures in Section 4.3 and 4.4.

A rollback priority is an arbitrary integer given by an operator. The management server rolls back the devices in order of this rollback priority.

Design Principles

The following two stages are executed.

1. Normal operation stage

The management server gets configuration files from all operational devices by executing all get procedures whenever an operator logs out from an operational device. The management server assigns an ID to a set of configuration files and hooked commands, along with backup time. Moreover, the management server requests the operator to enter an associable comment, in order to manage historical versions of operation logs and sets of device configuration. This stage can be performed on a regular basis and on demand from an operator.

2. Network-wide rollback stage

An operator sends a network-wide rollback request containing an arbitrary ID to the management server when the operator detects a network failure. The management server rolls back a set of configuration files to a past one that is associated with the ID, by executing rollback procedures in order of the rollback priority. If a new device is added after the time associated with the ID, there are many possible ideas for the rollback. One of the ideas is to make the management server roll back accurately by ignoring the added device. Because connectivity between the added device and other devices is lost by the network-wide rollback, the added device will have no effect on a network. On the other hand, if an operational device is deleted after the time associated with the ID, one possible idea is to make the management server alerts the operator that it is unable to roll back to the past one accurately. If the operator still requires the management server to roll back, the management server ignores the deleted device. Furthermore, if a device's firmware is updated, the management server assumes the device as a new device, because the format of the configuration file may be changed by the new firmware.

4.2 Overview of Operation Based on Proposed Scheme

Figure 3 shows an overview of operation based on the proposed scheme. When an operator performs a management task, he initially logs into the management server, and subsequently further logs into a target device from the management server, via an operator terminal (Fig.3 (S1) and (S2)). Every time the operator logs out from the device, the management server backups all device configuration files, using the get procedures (Fig.3 (S3)).

When an operator detects a network failure using an external network management system (Fig.3 (S4)), suspecting operator errors in the last management task (Fig.3 (S2)), the operator sends a network-wide rollback request to the management server (Fig.3 (S5)), whereupon the management server rolls back a set of configuration files to the past one in order of rollback priority. Thereby, the network is rolled back to a stable state at $T \leq t_0$ (Fig.3 (S6)).

Subsequently, an operator will conduct a cause analysis of the network failure. The proposed scheme supports comparison of the last set of configuration files at $T = t_1$ with other sets at $T \leq t_0$, and the verification of the commands hooked in the past

management task (Fig.3 (S7)). When the network failure is still alive, this might be due to deleted devices, or other aspects, such as a hardware failure. We may be able to cope with the deleted device and the hardware failure, by collaborating with virtualization technology, which is one of our future works.

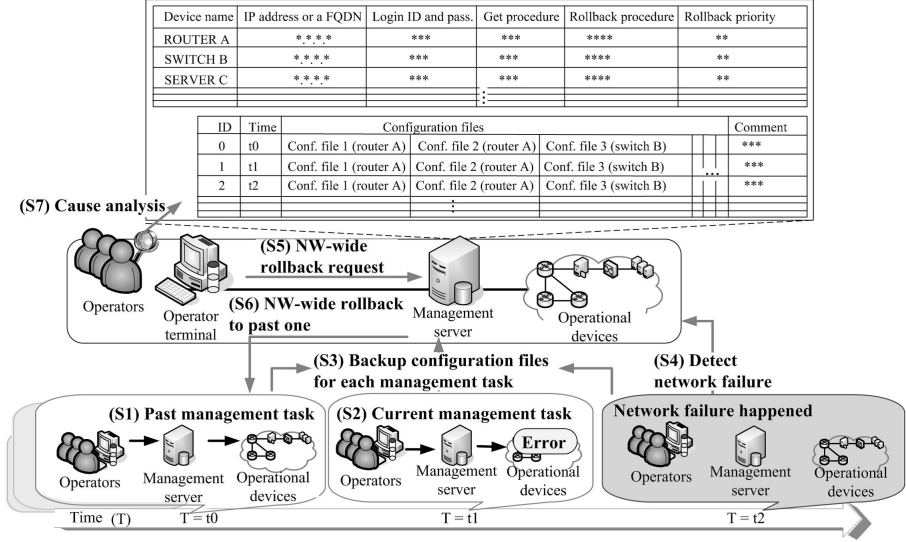


Fig. 3. Overview of operation based on proposed scheme

4.3 Details on Get Procedure

A get procedure is described in a small program such as shell script, and facilitates the management server getting configuration files from a device. An operator calls a get procedure, using a common protocol that is widely supported by devices, such as telnet and File Transfer Protocol (FTP). We show this procedure in more detail with an example of a Cisco router.

A typical Cisco router has two configuration files, namely a running-config file and a startup-config file [9]. These configuration files can be seen as a list of commands. The running-config file is always active on a device. A command which should affect device behavior changes the running-config file immediately. When a device is rebooted, it boots up with the startup-config file and sets up this startup-config file as a next new running-config file. The persistency of a command affecting device behavior cannot be guaranteed unless it is saved in the startup-config file.

When the management server runs a get procedure, the server takes the following three steps: (1) to establish a connection to a device via telnet with a login ID and a login password, (2) to get a running-config file via FTP to the local disk of the management server and (3) to disconnect the connection.

4.4 Details on Rollback Procedure

We provide the following three rollback procedures for fast recovery.

1. Full rollback procedure

The management server rolls back every configuration file of all devices. When the management server has received a network-wide rollback request, the server executes all rollback procedures in order of rollback priority. In the case of a Cisco router, the server takes the following three steps when executing a rollback procedure: (1) to establish a connection to a device, (2) to put a past running-config file to the router, and overwrite the existing startup-config file and (3) to reboot the Cisco router before logout.

2. Selective rollback procedure

The management server rolls back configuration files selectively. When the management server has received a network-wide rollback request, the server compares current and past configuration files on a device, as designated in the network-wide rollback request. If any difference emerges between them for each of the file pairs, the server executes a rollback procedure. The steps of a rollback procedure are the same as those of the full rollback procedure.

3. Command-based rollback procedure

Unlike the previous two procedures concerning the granularity of a configuration file, this procedure is that of a command. The management server automatically generates commands to rollback, and rolls back the configuration to the past one only by executing the commands without a device reboot. When the server has received a network-wide rollback request, the server conducts comparison of line-by-line between the current and past configuration files like the UNIX diff command, and automatically generates commands from the result of the comparison for every configuration file.

Figure 4 shows the detail of command generation in the case of a Cisco router. If some parameters in a configuration file such as “static route” were deleted from the current configuration file (Fig.4 (U1)), the difference of the configuration files will result in almost the same form as commands like “ip route 192.168.2.0 255.255.255.0 192.168.0.X” at the bottom-left of Fig.4, because these configuration files can be seen as a list of commands. On the other hand, if some parameters in a configuration file were added (Fig.4 (U2)), the concatenation of the word “no” followed by the difference of the configuration files like “no ...” at the bottom-right of Fig.4 would be a command to rollback.

By registering the procedures for each device configuration file, the proposed scheme performs in the multivendor network environment, and it can be deployed with no modification to all devices, thus it addresses Issue 1 and Issue 2. In addition, the proposed scheme considers the rollback sequence by the rollback priority, therefore it addresses Issue 3. Furthermore, the proposed scheme considers the addition and deletion of devices, such as by alerting, thus it addresses Issue 4.

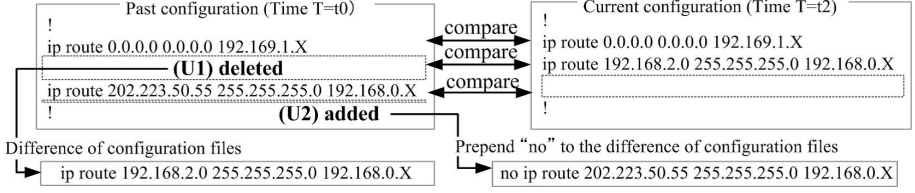


Fig. 4. Examples of command generation

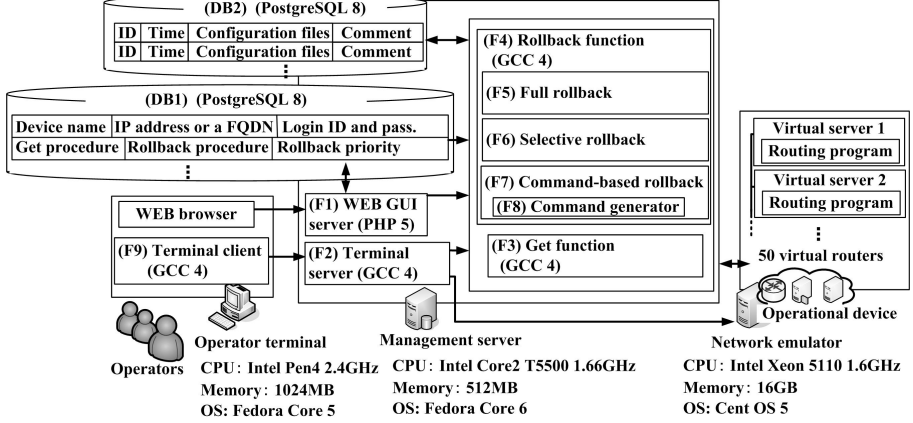


Fig. 5. Prototype system and measurement environment

5 Implementation

We implement a prototype system based on the proposed scheme to evaluate the time of the rollback. We describe a brief overview of the prototype system in Fig.5. We implement the management server, using a Linux server. The server has the following functions, (1) WEB GUI server (Fig.5 (F1)), (2) Terminal server (Fig.5 (F2)), (3) Get function (Fig.5 (F3)) and (4) Rollback function (Fig.5 (F4)). The rollback function has the following three sub functions, (4-1) Full rollback function (Fig.5 (F5)), (4-2)

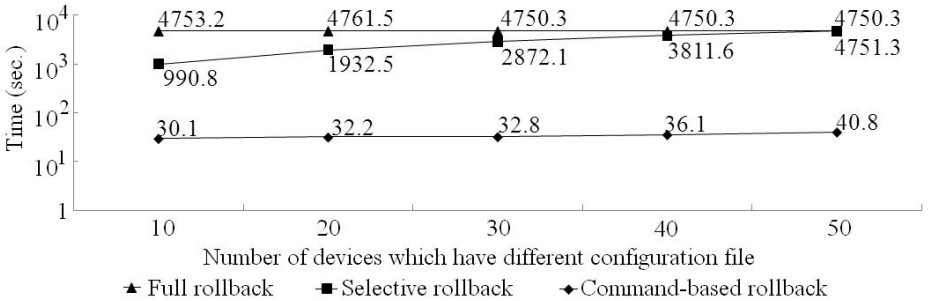


Fig. 6. Average rollback time of the three rollback procedures

Selective rollback function (Fig.5 (F6)) and (4-3) Command-based rollback function (Fig.5 (F7)). Each sub function corresponds to each rollback procedure shown in Section 4.4. The command-based rollback function has command generator (Fig.5 (F8)), which executes the steps of the command generation in Section 4.4. Additionally, the server implements two databases: one is for the preparation (Fig.5 (DB1)), and the other is for the management of the historical versions of operation logs and sets of device configuration. (Fig.5 (DB2)). We implement Terminal client (Fig.5 (F9)) on the operator terminal, which is a Linux box with a WEB browser, in order to provide a user interface for the operator.

6 Evaluations

We measure the rollback time by the three rollback procedures, using a prototype system. The bottom-right side of Fig. 5 shows the measurement environment.

A real network consists of many routers, and the average number of routers within a backbone network of a commercial ISP is approximately 32 [10]. To emulate such a real network, we install a network emulator, and configure 50 virtual routers using a virtualization technology [11] and a routing program [12]. Furthermore, since both full and selective rollback procedures require a router reboot, we set the reboot time of a virtual router to 95 seconds, which was based on the measurement of a small Cisco router (Cisco 1812j). If we compare the time of the command-based rollback procedure and that of the other two procedures, this reboot time is reasonable when we recall that high-end routers which have a longer reboot time are generally used within a backbone network. In the measurement, we change the number of devices with configuration files that differ between current and past ones: from 10 to 50 devices, for each 10 devices. Note that we set rollback priorities in order of the number of hops from the management server.

Figure 6 shows the average rollback time of the three rollback procedures. The full rollback procedure requires about 4750 seconds, regardless of the number of devices. The time of the selective rollback procedure is proportional to the number of devices. The rollback time of the command-based rollback procedure is less affected by the number of devices, and it can roll 50 virtual routers back in about 41 seconds.

We evaluate the proposed scheme in terms of MTTR. Some case study on a medium size regional network including 33 backbone routers reports that MTTR is in the order of about 20 minutes [13]. This report shows the total time of: (1) the time for the failure detection (Fig.2 (T1)), (2) the time for the cause analysis (Fig.2 (T2)) and (3) the time for the recovery operation (Fig.2 (T3)). If 75% of the time is spent just on detecting (Fig.2 (T1)) [6], the sum of the time of the cause analysis (Fig.2 (T2)) and the time of the recovery operation (Fig.2 (T3)) is in the order of about 5 minutes. This is longer than the time of the command-based procedure, thus the proposed scheme particularly with the command-based procedure can realize the dependable network by reducing MTTR.

Although the command-based rollback procedure may need implementation of command generator for every vendor and model, this will be alleviated by the activity such as IETF NETCONF working group that is standardizing common configuration file format.

7 Conclusions

This paper presents a detail of the network-wide rollback scheme for fast recovery from operator errors toward the dependable network. In the network-wide rollback, a network failure caused by operator errors is recovered by rolling the configuration files back to past ones. We showed four technical issues to realize network-wide rollback. Moreover, an overview of operation based on the proposed scheme was shown. We gave the following three rollback procedures: full, selective and command-based rollback procedure. We implemented a prototype system to evaluate the rollback time, and measured the time of rollback. The results showed the minimum rollback time was about 41 seconds, when 50 virtual routers were rolled back. We evaluated the proposed scheme in terms of MTTR. We showed that the proposed scheme would be able to reduce MTTR and realize the dependable network by reducing MTTR. Collaboration with virtualization technology to tackle deleted devices and a hardware failure will be our future work.

References

1. Patterson, D., et al.: Recovery Oriented Computing (ROC): Motivation, Definition, Techniques, and Case Studies. Computer Science Technical Report UCB/CSD-02-1175, U.C.Berkeley (2002)
2. Oppenheimer, D., Ganapathi, A., Patterson, D.A.: Why do Internet services fail, and what can be done about it? In: Proc. of USITS 2003 (2003)
3. International Electrotechnical Vocabulary Chapter 191 (May 2008), <http://www.electropedia.org/iev/iev.nsf/welcome?OpenForm>
4. Jeng, M., Siegel, H.J.: Design and Analysis of Dynamic Redundancy Networks. IEEE Trans. on Computers 37(9), 1019–1029 (1988)
5. Chen, M., Kiciman, E., Fratkin, E., Fox, A., Brewew, E.: Pinpoint: Problem Determination in Large, Dynamic Internet Services. In: Proc. IPDS Track 2002, pp. 595–604 (2002)
6. Kiciman, E., Fox, A.: Detecting Application- Level Failures in Component-based Internet Services. IEEE Transactions on Neural Networks 16(5), 1027–1041 (2005)
7. Brown, A.B., Patterson, D.A.: Undo for Operators: Building an Undoable E-mail Store. In: Proc. of USENIX 2003, pp. 1–14 (2003)
8. Yoshihara, K., Arai, D., Idoue, A., Horiuchi, H.: Proposal on Network-Wide Rollback Scheme for Fast Recovery from Operator Errors. In: Clemm, A., Granville, L.Z., Stadler, R. (eds.) DSOM 2007. LNCS, vol. 4785, pp. 199–202. Springer, Heidelberg (2007)
9. Cisco Systems Inc.: Cisco IOS Configuration Fundamentals. MacMillan Technical Publishing (1997)
10. CAIDA (May 2008), <http://www.caida.org>
11. Xen (May 2008), <http://xen.org/>
12. Quagga Routing Suite (May 2008), <http://www.quagga.net/>
13. Labovitz, C., Ahuja, A., Jahanian, F.: Experimental Study of Internet Stability and Backbone Failures. In: The 29th International Symposium on Fault-Tolerant Computing, pp. 278–285 (1999)

An Efficient Failure Recovery Scheme for Next Generation Network Services Based on Distributed Components

Wataru Miyazaki¹, Kazuhiko Kinoshita¹, Hideki Tode², Koso Murakami¹, Shinji Kikuchi³, Satoshi Tsuchiya³, Atsuji Sekiguchi³, and Tsuneo Katsuyama³

¹ Department of Information Networking, Osaka University, Japan
{miyazaki.wataru,kazuhiko,murakami}@ist.osaka-u.ac.jp

² Department of Computer Science and Intelligent Systems,
Osaka Prefecture University, Japan
tode@cs.osakafu-u.ac.jp

³ Organic Computing Lab., Fujitsu Laboratories Limited, Japan
{skikuchi,tty,sekia,KatsuyamaTsuneo}@jp.fujitsu.com

Abstract. With advances of information and communication technologies, many network services can be provided by assortment of desired components running on some nodes in the network, like a distributed component based service. It is effective to reduce the cost of service development. In addition, when a failure occurs, a distributed component based service has multiple components of each function, which can be used to replace the failed component. There is another problem, however a component failure affects the application servers using it. Therefore, in this paper, we propose a novel failure recovery scheme that constructs overlay networks between components having the same function and that provides an efficient search method for the substitute component using these multiple overlay networks. Finally, simulation experiments demonstrate the excellent performance of the proposed scheme.

1 Introduction

With advances of information and communication technologies, many network services can be provided by assortment of desired components running on some nodes in the network. It is effective to reduce the cost of service development. Moreover, in such an environment, since the workload can be distributed over multiple components, the scalability of the service can be improved. As a typical example, distributed web service technology has been under the spotlight [1]. In this paper, we call such a service *distributed component based service*. It is expected to be standard for next generation network services [2], [3].

On the other hand, when a failure occurs on a service, traditional network services by single server can no longer provide the service until the failure is recovered. By contrast, a distributed component based service has multiple components of each function, which can be used to replace the failed component.

However, distributed component based services have another problem: a component failure affects the application servers using it. Hence, to restart the service, such application servers must search for an alternative component of the same function on the network, and instantly switch from the failed component to another sound substitute component. It is necessary that the target service retains its quality after substituting the failed component with an operational one. Therefore, in this paper, we discuss a way of replacing each failed component into the corresponding suitable substitute component when a failure has occurred in the distributed component based service.

Search methods for distributed component based services can be categorized as either proactive or reactive.

A major part of proactive searches is UDDI (Universal Description, Discovery and Integration), which is defined as a kind of registry and used for searching the current web service [4]. A user can obtain the requested web service address by searching for the target web service in the UDDI registry with keywords, which are matched against the registered keywords. In addition, some extended methods using UDDI have been introduced [5]-[8]. These methods can immediately find a component offering high-quality service to the users.

However, the method is not appropriate for searching for substitute components when a failure has happened, because of centralized management. When a failure occurs, the failure information does not reach the central registry immediately. Therefore, for example, when an application server searches for a substitute component for the failed component by the UDDI directory, the directory passes on the address of the component that had the best performance before the failure occurred, which may be the address of the failed component. In addition, a central control registry can be a single point of failure.

On the other hand, flooding that is one of reactive searches provides a simple way to overcome this problem [9]. In such a method, queries are broadcast to all components near the application server when a failure occurs. After that, the application server changes the failed component to the component replying to the broadcast message. Therefore, substitute components can be found regardless of the scale of the network. Wherever a failure happens, this method can find an appropriate substitute component. In other words, the method is fault tolerant. However, it is difficult to find a substitute component quickly because we have no information prior to the failure. Furthermore, the flooding leads to the increase in traffic after a failure.

Consequently, we need a scheme for finding a suitable substitute component quickly without a massive increase in traffic. Therefore, we propose a novel failure recovery scheme that constructs overlay networks between components having the same function and that provides an efficient search method for the substitute component using these multiple overlay networks. Overlay networks can be reconstructed flexibly, so that the increase in traffic can be prevented by limiting the number of links or that quick finding of the substitute component by increasing of the overlay degree of each node. It easily adapts to change

of environment, e.g. new components added to or removed from the network, performance upgrade of nodes, and so on.

2 Model of Distributed Component Based Service

In this section, we first describe a model of a distributed component based service and the problematic issues. Fig. 1 shows that three types of components a , b and c are running on six nodes, and an application server α provides network services by connecting components a_4 , b_1 and c_1 . If, for example, when component c_1 on node 1 fails, the application server α searches for another component that has the same function in the network, and finds component c_3 in this Fig. 1. It restarts the service using it as a substitute for component c_1 .

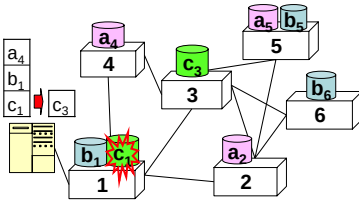


Fig. 1. Model of distributed component based service

component, and so on. In this paper, we consider TAT (*Turn Around Time*) as a function including these elements.

In our proposed scheme, TAT is defined as a weighted sum of the following two functions. One is a function that indicates the processing time of a component, and the other reflects the delay of links incurred by traffic from the application server to the node where the component is running.

At first, we define p_i , u_i , b_a and PT_{ai} as the processing power of node i , the number of accesses per unit time to all components running on the node i , the throughput of component a that is the processing power to finish a process till the unit time, and the processing time till finishing a process by performance of component a_i , respectively. The processing power per access is obtained by dividing the processing power of node, p_i by the number of accesses on the node, u_i . The time PT_{ai} for component a_i to finish a process is obtained by dividing the throughput of component, b_a by the processing power of one access. Next, we define RTT_{ai} as the performance of links, given by RTT from α to node i having component a_i .

Finally, we define the performance evaluation function TAT_{ai} , given by (1).

$$TAT_{ai} = \omega / PT_{ai} + (1 - \omega) \times RTT_{ai} \quad (1)$$

TAT_{ai} is calculated by summing the previous two functions with a weight ω . Unless otherwise specified, “ TAT ” in this paper means TAT_{ai} . The weight ω can be varied depending on the type of the provided services, for example, if the process of service is light, the weight ω becomes small.

Quickly to find a substitute component that provides smaller TAT, we have to consider two factors: how to determine the location where replicated components with the same functionality run and how to search for a substitute component after a failure occurs. However, it is difficult to decide the best place for running the replicated components, because a new replicated component has to be aware of components already running on the network or components that may have to be relocated to new nodes. It is impossible to track the condition of all the components on a large-scale network. Therefore, in this paper, we do not suppose a specific method of component placing, in other words, we assume that each component including replications runs on a randomly selected node.

The above scenario involves just one failed component. However, more often an entire node fails and all components running on that node become disabled. Then all applications using these disabled components have to find substitute components simultaneously. In general, a component may be used by two or more applications, so even when a single component fails, several applications using the failed component may start searching for the substitute component simultaneously. It is necessary not only to realize rapid search but also to prevent too large increase in traffic because of the many concurrent searches. Furthermore, unless the method uses the dynamically changing information of TAT, the substitute component will be chosen based on information created prior to the failure. Consequently, substitute components may be concentrated accidentally on a single node, leading to a load imbalance and the failure of the overloaded node. Our proposed scheme tackles such issues.

3 Proposed Scheme for Finding Substitute Components

3.1 Overview

In this paper, we propose constructing multiple overlay networks in order to find substitute components efficiently. It is desirable for finding a suitable substitute component that each component perceives performance information of all components, such as processing power, the number of accesses and so on. In such a manner, however, there are too much traffic on the network and each component has to deal with too much information.

It is not feasible from the viewpoint of scalability. Overlay networks define the range of information management. In addition, by constructing overlay networks per component, each component does not receive messages from other kinds of components. In the proposed scheme, the total number of connected links of each component in the overlay network is limited. These features prevent flooded traffic in the search method from becoming excessive. Moreover, even if an overlay

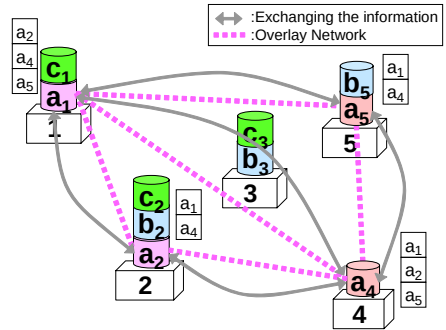


Fig. 2. Steady state phase

network topology is changed, other overlay networks are not changed. It achieves a flexible management for each component.

Our proposed method has the following three phases; constructing phase, steady-state phase and failure-recovery phase.

Constructing phase. Overlay networks are constructed per a kind of components. This phase is invoked whenever one or more components are added. In addition, each component defines *searching component*, which is asked for a suitable substitute component from the application server when a failure occurred. It is selected from its adjacent components and has the most connected links among them. Subsection 3.3 explains this phase in more detail.

Steady-state phase. While a distributed component based service is provided, each component exchanges its performance information with adjacent components on its overlay network. For example, in Fig. 2, components directly connected on the overlay network, like components a_4 and a_5 , exchange performance information, and the exchanged information is recorded as a table on each node. Therefore, despite implementing the same function, components a_2 and a_5 , that are not connected directly on the overlay network, cannot exchange performance information. This is for reduction of processing load of each node and traffic all over the network.

Failure-recovery phase. This phase starts when an application server is notified of a component failure. The searching component asked to detect a substitute selects most suitable component for the failed component from among the candidates for substitute. The selection depends on TAT calculated before the failure occurred. Subsection 3.2 explains this phase in more detail.

3.2 Searching for the Substitute Component

During the failure-recovery phase, the application server using the failed component asks the searching component to search for a substitute component. The searching component calculates TAT based on information on the performance of the node and the RTT recorded in its own table. Next, the searching component selects the components whose TATs are the smallest and second-smallest. The searching component sends a message to these components. At the time of receiving a message, the component calculates its own TAT, and compares it with TAT included in the message. After this comparison, the component sends the searching component a reply message with the information needed to calculate the component's new TAT. Specifically, if smaller than the second-smallest TAT included in a message, the component adds a permission-to-use flag to the reply message. At the time of receiving the reply message, the searching component updates the information of the replying component in its own registry. If the reply message includes the permission-to-use flag, the searching component transmits the component's address to the application server. If the reply message does not accept the request (i.e., lacks a permission-to-use flag), the searching component recalculates the component's TAT with the updated information in

its own table. These processes continue until a candidate for substitution gives its permission. After that, the application server can use the component with the smallest TAT among the candidates for substitution.

3.3 Constructing Multiple Overlay Networks

In the proposed method for searching for substitute components, only the searching component and the components connected to the searching component on the overlay network can be candidates for the substitution. In other words, the more components the searching component can compare if more components are connected to it on the overlay network, the easier it will be for the searching component to select the component that has the smallest TAT. However, the more components the component is connected to in the overlay network, the more information it will have to process for calculating TATs. Furthermore, in this method, the components that are interconnected on the overlay network regularly exchange information to calculate TAT of each component, and as a result, as the number of connected links increases, so does the regular traffic. For these reasons, the topology of the overlay network is one of the most important elements affecting the performance of searches for substitute components.

From the viewpoint of managing large scale-networks, we decide to limit the total number of overlay links over all components running on a node. This limit value M_i depends on the processing power of the node i . Then, we focus on components with different functionalities on the same node. The total number of links should be distributed as fairly as possible across each component running on the node. From these two considerations, our proposed method defines an overlay network topology that optimizes effectiveness. We explain how to construct overlay networks in detail using the variables as shown in Table 1.

Table 1. Variable definition

variable	definition
x_i	component on the node i
d_{xi}	number of connected links to the component x_i
D_i	total number of connected links on the node i
M_i	upper limit of the total number of connected links of the node i
L	lower limit of connected links per component
$\bar{d}_i$	average number of connected links of all components running on the node i
C_{1i}	set of components running on the node i
C_{2i}	set of components connected to the component x_i included in C_{1i}

Specifically, during the constructing phase, the proposed method constructs overlay networks with components of the same functionality like Fig. 3. First, we focus on node i , and define x_i , d_{xi} , and D_i as the Table 1. For example, in the case of node 2 in our previous example, the number of connected links d_{a2} , d_{b2} and d_{c2} of components a_2 , b_2 and c_2 is two for all of them. D_2 , the total number of connected links of node 2, is six ($= d_{a2} + d_{b2} + d_{c2}$). Let M_i denote

the upper limit of the total number of connected links of node i . This value is predetermined based on the processing power of node i , to avoid overloading.

By contrast, from the viewpoint of search efficiency and fault tolerance, it is desirable to have more links to other components. It is especially important that the overlay network is not fragmented by the failure of a component or node. Thus, we need to set a lower limit L —a fixed number that is independent of processing power—and components must be connected to others of the same functionality so as to exceed lower limit L unless the number of connected links exceeds L . The value L is set to 2 in our proposed method for simplicity. Fig. 4 is the flowchart that shows how new links are created when a new component is added during the constructing phase or the steady phase.

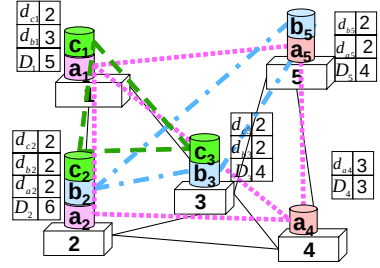


Fig. 3. Multiple overlay networks

Fig. 4 is the flowchart that shows how new links are created when a new component is added during the constructing phase or the steady phase.

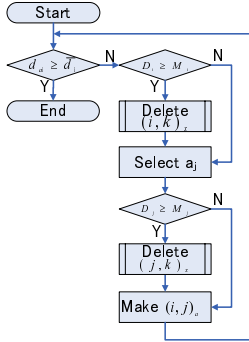


Fig. 4. Flowchart of constructing an over-

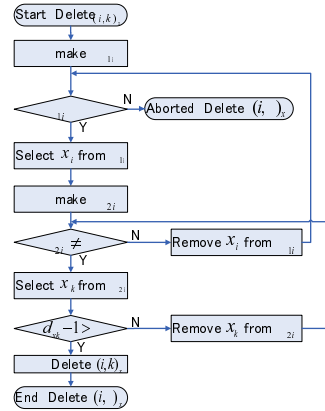


Fig. 5. Flowchart of deletion of existing links

The newly added component a_i makes new links until the number of connected links d_{ai} reaches $\bar{d}_i$, which is the average number of connected links of all components running on the node i , for making homogeneous load of each component. This value is obtained from the upper limit of the number of connected links of node i divided by the number of components on the node i (including the component a_i). As a result, the component a_i can have as many candidates for substitution as any other component on node i . After node i broadcasts a connection request message including the ID of component a_i to the components a , the component a_i waits for the replies. Finally, the component a_i tries to connect a new link to the component a_j . The order of connection is the order of reply arrivals.

To prevent D_i from becoming greater than M_i , node i deletes an existing link. Fig. 5 is the flowchart of this process. Let C_{1i} denote the set of components running

on the node i . First, a component x_i with the largest number of connected links in C_{1i} is selected. Second, let C_{2i} denote a set of components connected to a component x_i on the overlay network. Third, the component x_k with the largest number of connected links in C_{2i} is selected. Finally, the link $(i, k)_x$ between component x_i and component x_k is deleted.

However, if $d_{xk} - 1 < L$, executing the link deletion process will result in too few links on node k , so the component x_k is removed from C_{2i} , and a component $x_{k'}$ with the largest number of connected links in C_{2i} is selected. If C_{2i} is empty because the process of removing the component x_k from C_{2i} is repeated too many times, component x_i is removed from C_{1i} , and a component x'_i with the largest number of connected links in C_{1i} is selected. If C_{1i} is empty because the process of removing the component x_i from C_{1i} is repeated too many times, the process of deleting an existing link is aborted, and the process of connecting a new link to component a_i is finished.

After an existing connected link is deleted by this recursive process, the total number of connected links D_j is compared with the upper limit of the total connected links M_j . If $D_j \geq M_j$, then node j decides to delete one of its existing links. In this case, if the process of deleting an existing link is aborted, component a_i decides not to connect a link between the components a_i and a_j . After that, component a_i tries to connect a new link between the components a_i and $a_{j'}$ for $j' \neq j$. In this way, new links of component a_i are connected in a way that prevents the total connected links per node j from reaching the upper limit of total connected links. Through these processes, each overlay network that includes components with only the same functionality is constructed. In these multiple overlay networks, the load is evenly distributed across nodes and the number of connected links of each component on a node is almost equal.

In addition, if the number of connected links of all components is equal to the lower limit L , no component can make any further links. This is because there are too many components running on the network for acceptable performance. We exclude such a case from this paper.

4 Performance Evaluation

4.1 Simulation Model

To evaluate our proposed method, we simulated it under the following conditions.

A network based on a Barabasi–Albert scale-free network [10] with 100 nodes was constructed. The delay between each node was 1 (unit of time) for simplicity. Ten types of components were prepared, and their throughputs were of constant value: 2.0×10^{-2} . The total number of accesses to each type of component per unit time was set to a constant 300. The number of accesses per unit time on an application server is 100. The processing power p_i of node i are $0.2, \dots, 1.0$ at each 20 nodes and the upper limit of total connected links is the value, $100 \times p_i$.

Two methods were used for comparison. One was the method named First Reply. With this method, after the flooding messages were broadcast to search for the substitute component, the component that returned the fastest reply

among all the components with the same function was selected. The other was the method named Best Reply. With this method, after the flooding messages were broadcast to all areas of the network to search for the substitute component, the component that had the smallest TAT of all the components was used as a substitute component.

4.2 Simulation Results

Fig. 6 shows TAT of the substitute component when a node failure occurred in the case of $\omega = 0.5$. Also, Fig. 7 shows the time between starting searching for a substitute component and finding one in the same case. The x-axis of all graphs shows the number of components of the same function on the network.

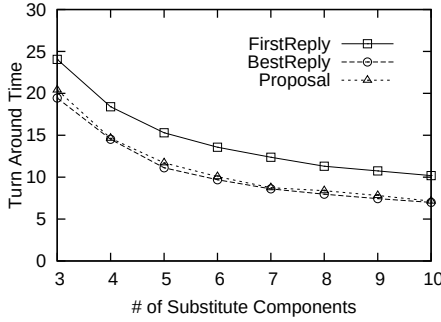


Fig. 6. Turn Around Time ($\omega=0.5$)

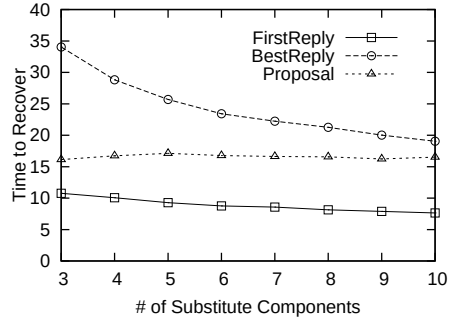


Fig. 7. Time to Recover ($\omega=0.5$)

First of all, Fig. 6 shows that the proposed method can find a substitute component that has as high performance as Best Reply can in spite of the overhead of check function. While, Fig. 7 shows that Proposal takes longer time to recover than First Reply but attains the performance just between First Reply and Best Reply. Next, as shown in Fig. 8 and 9, independently of the weight ω , the proposed method can consistently search for a substitute component whose processing speed is fast as well as Best Reply.

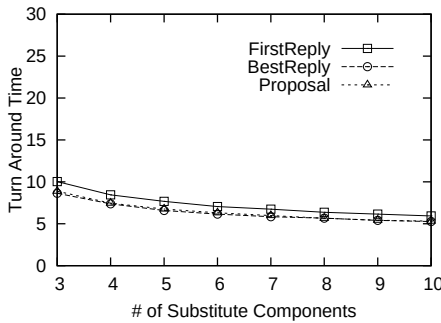


Fig. 8. Turn Around Time ($\omega=0.1$)

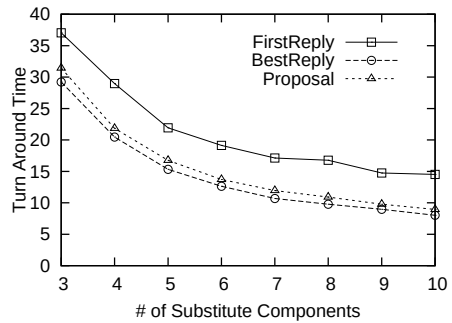


Fig. 9. Turn Around Time ($\omega=0.9$)

In addition, we conducted a simulation experiment where the number of nodes of a specific processing power was especially large, for example, where the number of nodes with low processing power is large or where the number of nodes with medium processing power is large. Even in such cases, there was no specific performance difference though we omitted the results due to page limitation. From those results, our proposed method can find a more suitable substitute component than First Reply and find more quickly than Best Reply.

5 Conclusion

In this paper, we proposed an efficient failure recovery scheme for next generation network services. It achieves to search for an adequate substitute component quickly, when a failure occurs at a node on a distributed component based web service. The proposed scheme constructs component-level multiple overlay networks as a platform for substitute component searching. The increase in traffic can be prevented by limiting the number of links and the network adapts to addition/removal of components. Finally, we confirmed that the search method using multiple overlay networks was effective on node failure, and that the proposed scheme adapted to various situations of network.

Future work includes dynamic reconfiguration of overlay network topologies following updated information of TAT, which is unavailable at the first configuration to make the proposed method to detect desired components more rapidly. As a result, the application server can reselect the component accordingly.

References

1. Weerawarana, S., Curbera, F., Leymann, F., Storey, T., Ferguson, D.F.: Web Services Platform Architecture: SOAP, WSDL, WS-Policy, WS-Addressing, WS-BPEL, WS-Reliable Messaging, and More. Prentice Hall, Englewood Cliffs (2005)
2. Lee, C., Knight, D.: Realization of the next-generation network. *IEEE Communications Magazine* 43, 34–41 (2005)
3. Falcarin, P., Licciardi, C.A.: Technologies and Guidelines for Service Creation in NGN, Technical-scientific Magazine of Telecom Italia Lab Researchers (December 2003)
4. UDDI V3.0.2, http://uddi.org/pubs/uddi_v3.htm
5. Al-Masri, E., Mahmoud, Q.H.: A Framework for Efficient Discovery of Web Services across Heterogeneous Registries. In: 4th IEEE Consumer Communication and Networking Conference (CCNC 2007), pp. 415–419 (January 2007)
6. Al-Masri, E., Mahmoud, Q.H.: Discovering the best web service. In: Proceedings of the 16th international conference on World Wide Web, pp. 1257–1258 (May 2007)
7. Ran, S.: A model for web service discovery with QoS. *ACM SIGecom Exchanges* 4, 1–10 (2003)
8. Hao, Y., Zhang, Y.: Web services discovery based on schema matching. In: ACM International Conference Proceeding Series, vol. 244, pp. 107–113 (January 2007)
9. Tode, H., Wada, M., Kinoshita, K., Masaki, T., Murakami, K.: Enhanced Flooding Algorithms Introducing the Concept of Biotic Growth. *IEICE Trans. Commun., Special Section on Ubiquitous Networks* E88-B(3), 903–910 (2005)
10. Faloutsos, M., Faloutsos, P., Faloutsos, C.: On power-law relationships of the Internet topology. *ACM SIGCOMM Computer Communication Review* 29, 251–263 (1999)

A Novel Fault Diagnosis Approach to Path-Protected WDM Mesh Networks

Chi-Shih Chao

Department of Communications Engineering
Feng Chia University, Taiwan 40724, ROC
cschao@fcu.edu.tw

Abstract. In today's cyber-world, network faults can not only cause unexpected damages and consternation to enterprises, but also result in huge loss of customers and a lot of negative effects. For this reason, ISPs (Internet Service Providers) do their best to achieve the goal of higher network reliability, service availability, and accessible bandwidth. To do so, most of them construct or rent carrier-grade backbone networks, which use WDM (Wavelength Division Multiplexing) with path restoration/protection techniques, as one of the means toward the goal. However, network faults can still impact largely to these well-protected WDM networks since the actual network reliability would fall down as a fault occurs, even if the network could function as normal with protection mechanisms. In addition, the protection/restoration mechanisms would usually get fault diagnosis into trouble because they can cover faults in most cases. In this paper, a novel and competent fault diagnosis approach which can be used in path-protected WDM mesh networks for a high and constant network availability is proposed. This diagnosis approach uses the trimmed/extracted major alarm propagation behavior of a fault and our leader major alarm domain concept to deduce and isolate the most suspicious network area of the fault. Additionally, to enhance the diagnosis accuracy, our approach also refers to some certain information with respect to SRLG (Shared Risk Link Group) and dynamic network sessions of the diagnosed network to further narrow down the suspicious area. Last, simulation results associated with discussions are shown to demonstrate the performance of our diagnosis approach.

Keywords: Network fault management, WDM mesh networks, SRLG, trimmed major alarm propagation pattern, leader major alarm domain.

1 Introduction

In recent years, due to the emergence of WDM and associated protection/restoration technologies, networks have the ability to carry huge amount of data on them, with high availability and reliability [1]. As a result, it can be seen that these techniques are widely utilized or deployed within core/metro backbone networks. However, faults can still affect such a well-protected and highly-performed WDM network because they can greatly reduce the overall availability/reliability of the network and its associated services, even if users over the network could feel no harms occur at all

[2, 3]. A number of activities engaging in fault management for WDM networks, which range from theoretical to practical topics, have been ongoing for years [3-9]. Among these activities, the research results coming from Carmen Mas et al. [5, 6] are generally viewed as being the most noticeable and feasible. Carmen Mas et al. propose their fault localization approach (FLA), which is developed on the European Continental WDM Mesh Network, to detecting and localizing faults. In their approach, they deploy and configure optical measurement equipments (e.g., power-meter, ONSR/BER monitor, etc.) all over the diagnosed WDM network. Later, they collect alarms which are derived from the measurement readings of those equipments, and then use a codebook-like tree-based scheme to perform alarm correlation for fault isolation [12]. Although they announce their approach performs quite well on the basis of their simulation results, however, their work doesn't consider effects on fault isolation which are resulted from the involvement of some critical items in reality during the construction of WDM networks; e.g., path protection/restoration mechanisms, SRLG (Shared Risk Link Group), and so on [1]. Besides, fault isolation in such a way should think over the deployment problem of the measurement equipments (or probes) – it has been proved as an NP-complete problem [10]. These would prevent their current research results from being at the stage of real deployment.

On the basis of historical statistics which show fiber cut has a dominating major among faults in WDM networks [11], in this paper, we put our focus on effectively isolating the suspicious faulty area in path-protected WDM mesh networks with the occurrence of only a single fiber cut. To let the work be well done, the approach we propose utilizes the trimmed/extracted major alarm propagation behavior of a fiber cut as well as the leader major alarm domain concept for the isolation of the fault. Our approach also includes some certain information coming from SRLG (Shared Risk Link Group) and dynamic network sessions on the diagnosed network to further narrow down the suspicious area effectively. Simulation results demonstrate that our approach is up to the task. The rest of this paper is organized as follows. In Section 2, alarms collected are classified into two groups: major and minor alarms to reduce the amount of alarms needed to be analyzed for fault diagnoses. Then, the concept of a trimmed major alarm propagation pattern which can effectively reveal what a fiber cut truly behaves like within a protected WDM network is described. Based on this concept, Section 3 further proposes and utilizes the leader major alarm domain concept to keep narrowing down the suspicious faulty area for inspection. Plus, we also add some certain information, which is retrieved from SRLGs and dynamic light sessions of the diagnosed network, into our fault for more precise isolation. Section 4 shows our simulation results with some useful discussions to demonstrate the performance of our approach and Section 5 concludes this paper with future works.

2 Alarm Propagation in Path-Protected WDM Mesh Networks

To develop a competent system (or approach) of automated fault isolation in a path-protected WDM mesh network, in essence, it is necessary to clarify the fault behavior first. The term “fault,” in general, refers to the malfunction or failure of a component that causes the whole network to deviate from its normal behavior. Basically, a fault

can reveal itself via a number of alarms (or symptoms) that it causes. However, there could be hundreds or thousands of alarms appearing when a fault occurs and this makes it quite difficult to know what the fault is. Thus, at the very beginning, we undertake specifying precisely the behavioral pattern of a fault in a path-protected WDM mesh network, where a fault here means a fiber cut.

2.1 Fault Behavior Extraction and Detection

When a fiber cut occurs in a path-protected WDM network, for example in Fig. 1, the light intensity perceived by the optical receiver of the communication device OXC2 would fall down below the designated lower threshold. In this case, the OXC2 will emit a “No Signal” alarm to the network management system via the control plane network of a diagnosed WDM network [12] because it can retrieve no signal (or data). For the Router B in the same case, although it can perceive the light ray with normal intensity regenerated by OXC2, it will still emit “No Signal” since OXC2 can restore the light power but not the original signal. By the same token, devices which are further down the λ_1 's light session path are supposed to send out the same “No Signal” alarm as Router B does. However, due to the establishment of the backup path labeled λ_2 , “No Signal” alarms from those devices which are along the λ_1 's light session path but behind Router B might not show up, depending on the time needed to establish the backup path. Such nondeterministic behavior in a path-protected WDM mesh network implies that only those alarms with constant appearance for the occurrence of a fault can assist us in the precise behavior description of that fault.

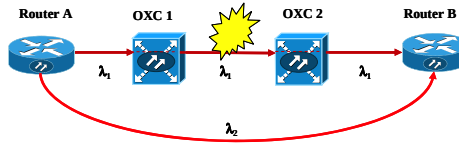


Fig. 1. Path Protection in a WDM Network

While modeling/specifying the fault behavior using alarms, one another thing worth being noticed is: The network management system should receive different categories and amounts of alarms by varying the fault occurrence location and time in the monitored DWDM mesh network [13]. Thus, in our work, for precise specification of the fault pattern in path-protected WDM mesh networks, alarm dependence relationship [14] is utilized to classify alarms based on their “importance” for the corresponding fault into two major groups: (1) *major alarms* mean they can always appear whenever and wherever a fault occurs, ex., the “No Signal” alarm is the major alarm of a fiber cut; (2) *minor alarms* may or may not appear when a fault occurs, ex., the “Incomplete Burst” alarm, which indicates a device does not receive the complete data burst, can be one of the minor alarms of a fiber cut. By this classification, we can filter minor alarms out of the alarm collection database to effectively analyze/obtain the fault behavior, and hence improve the management system’s feasibility and scalability. As a result, a fiber cut can be described or modeled as a trimmed but more precise string of “No Signal” alarms which

propagate, along the session path, from the optical device right behind the fault location to the device ending (or connecting) the backup path. Figure 2 shows the modeled fault behavior with the complete system architecture of our approach.

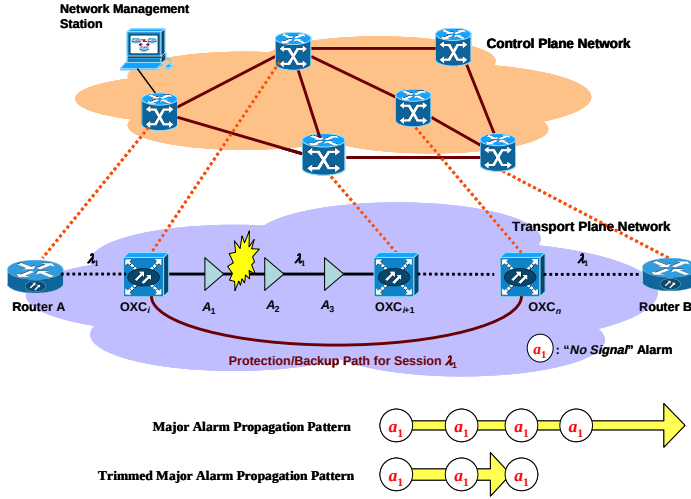


Fig. 2. Fault Behavior Modeling with Trimmed Major Alarm Propagation

2.2 Leader Major Alarm Domain

Nevertheless, fault diagnosis depending only on the trimmed major alarm propagation pattern can not satisfy demands of our fault isolation over a large-scale WDM mesh network. Based on the alarm domain theory [15], in the case of the fiber cut in Fig. 3(a), network components which have the possibility to cause the appearance of the “No Signal” alarms from OXC2 and Router B can be treated as suspects to fail, i.e., all of the constituent optical devices and elements in-between OXC1 and Router B. If we go deep into Fig. 3(a), it can be found that there actually exist much more devices/ elements in-between OXC1 and Router B than it may imply (as shown in Fig. 3(b)), which may be spaced from several to hundreds of miles away between each other. If they all are thought of as our final diagnostic results, the time and cost to check them one by one for fault diagnoses should not be allowable.

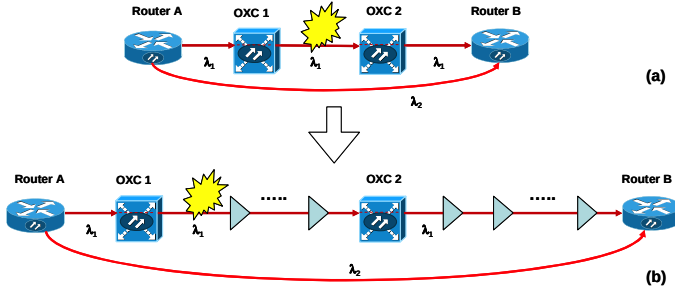


Fig. 3. Light Session Topology

For this problem, our leader major alarm domain concept is employed [14]. In Fig. 4, there would be a_1 : “No Signal” alarms coming from OXC_{i+1} to OXC_n , and Router B when the fiber cut occurs. Since the first alarm in the alarm propagation pattern can reveal the beginning time and the most possible location of a fault, in Fig. 4, we use the “No Signal” alarm from OXC_{i+1} to supplant the whole trimmed a_1 alarm string for fault diagnoses. Then, by employing the alarm domain theory [13], we can deduce a smaller suspicious element set for the OXC_{i+1} ’s a_1 : $\text{Domain}(\text{Alarm}(a_1, OXC_{i+1})) = \{OXC_i, \text{Link}(OXC_i-A_1), A_1, \text{Link}(A_1-A_2), 3A_2, \text{Link}(A_2-A_3), A_3, \text{Link}(A_3-OXC_{i+1}), OXC_{i+1}\}$, totally 9 elements. By contrast, if our leader alarm domain concept is not used, all devices and elements in-between OXC_{i+1} and Router B would be regarded as suspects. Note that A_i here stands for the i th optical amplifier. In opaque optical transport networks, 3R (Reshaping+Retiming+Regeneration) can be employed to perform the same signal amplification.

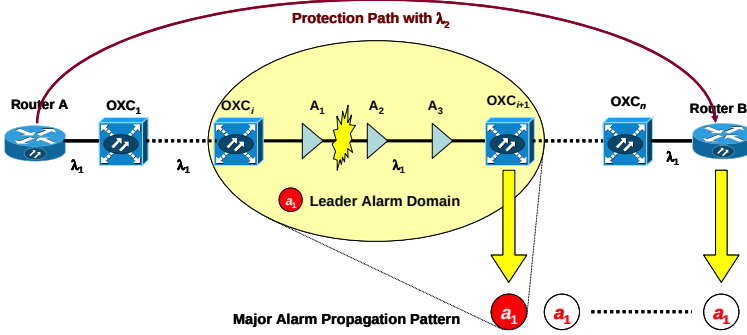


Fig. 4. Leader Major Alarm Domain

3 Fault Isolation

In our work, the diagnostic results at the end of Section 2 are still far from being adequate as the final version. The reasons are:

- (1) Any two of the elements in the suspicious set could still be placed far apart, especially in large-scale WDM mesh networks [5]. In practice, the physical distance between points A_1 and A_2 in Fig. 4 can be several to hundreds of kilometers away.
- (2) In addition, the suspicious element set is not that “precise,” actually. For example, if Fig. 4 is drawn in more detail as Fig. 5, then diagnostic results should be $\{\text{Switch-Fabric}(OXC_i), \text{Transmitter}(OXC_i, \text{Fiber}_N), \text{MUX}(OXC_i, \text{Fiber}_N), \text{Amplifier}(OXC_i, \text{Fiber}_N), \text{Link}(OXC_i-A_1), A_1, \text{Link}(A_1-A_2), A_2, \text{Link}(A_2-A_3), A_3, \text{Link}(A_3-OXC_{i+1}), \text{Amplifier}(OXC_{i+1}, \text{Fiber}_N), \text{DEMUX}(OXC_{i+1}, \text{Fiber}_N), \text{Receiver}(OXC_{i+1}, \text{Fiber}_N), \text{Switch-Fabric}(OXC_{i+1})\}$, where the number of suspects is augmented from 9 elements mentioned at Section 2.2 up to 15 elements. For a single occurred fault, this is definitely not a good diagnosis with 15 elements to look forward to being checked or inspected one by one. In words, (1) and (2) imply that we need additional and more useful information as evidence to further narrow down the number of suspicious elements effectively.

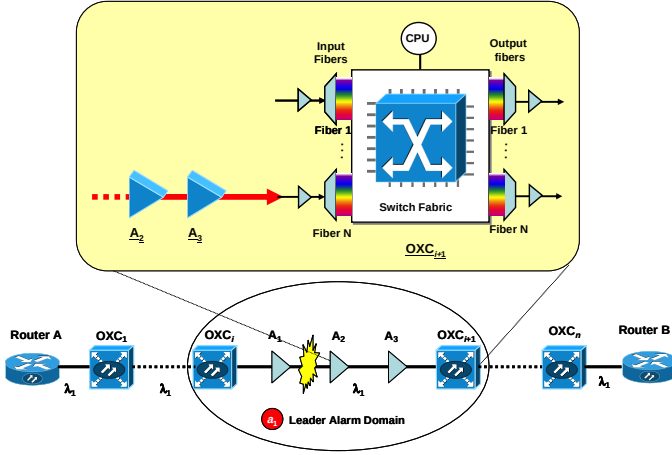


Fig. 5. A Zoom-ined Light Path Section

3.1 Healthy/Positive Information for Fault Diagnosis

To further enhance our fault diagnoses, some other information should be considered and included in the process of fault isolation to downsize the suspicious element set. The first type of such information which can facilitate this work is the dynamic light session information in the diagnosed WDM mesh network while faults occur. This type of information indicates that those devices/elements, which are within the territory of the leader major alarm domain and meantime are traveled through by one of the workable light sessions during the period of faults, should be treated as the profitable information source (or evidence) for fault diagnosis. Since these devices/elements can function well in the diagnosed network when a fault occurs, we should remove them from the suspicious element set of that fault. In the case of Fig. 6, during the period of the fiber cut which occurs between A_1 and A_2 , two sessions, $Session_{i+1}$ and $Session_{i+2}$, remain normal (e.g., normal session establishment or data transmission). Accordingly, we have to eliminate the elements Switch-Fabric(OXC_i) and Switch-Fabric(OXC_{i+1}) through which these two sessions pass from our deduced diagnosis results.

In addition to the information derived from “healthy” (i.e., “normal”) dynamic optical sessions, fault diagnosis in WDM mesh networks should also consider the physical network construction (or physical layout). In our work, SRLGs (Shared Risk Link Groups) for the physical construction of WDM mesh networks are taken into account. For some reasons (e.g., cost, administration, geography, environment, or else), it can be found that two or more sessions share the same conduit/PVC pipeline/cable at some link sections [1] in most of cases, where these sessions are viewed as being within the same SRLG. As shown at the bottom half in Fig. 6, $Session_i$ and $Session_{i+3}$ are in the same SRLG for some conduit. If $Session_{i+3}$ can work normally during the period of the fiber cut, those elements within the SRLG should be treated as “healthy” or “innocent” for fault diagnosis. In the case of Fig. 6, by employing the concept of SRLG, we should further delete those communication components through which $Session_{i+3}$ passes from the suspicious element set; they are A_2 , Link(A_2 - A_3), A_3 , Link(A_3 - OXC_{i+1}), Amplifier(OXC_{i+1} , Fiber $_N$), DEMUX(OXC_{i+1} , Fiber $_N$), Receiver(OXC_{i+1} , Fiber $_N$).

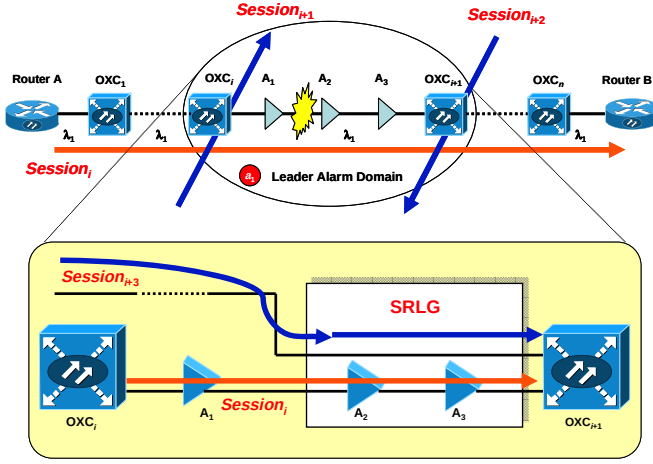


Fig. 6. Healthy/Positive Diagnosis Information

As a result, by considering the “healthy/positive diagnosis information” which comes from dynamic light sessions as well as SRLGs into fault isolation, we can deduce 6 most suspicious elements: {Transmitter(OXC_i , Fiber $_N$), MUX(OXC_i , Fiber $_N$), Amplifier(OXC_i , Fiber $_N$), Link(OXC_i - A_1), A_1 , Link(A_1 - A_2)}. It is only 40% of the suspicious elements in diagnosis results which are deduced at the end of Section 2.2 by merely taking the leader major alarm domain concept to reason.

4 Performance Evaluation and Discussions

To evaluate the fault isolation performance of our approach, the SRLG-embedded topology of the path-protected WDM mesh network managed by the Taiwan High Speed Rail Co. (THSR) is used. The Immediate-Setup-with-Explicit-Release scheme [1] is employed as the signaling protocol for the establishment and release of light sessions, and the 1+1 preplanned path protection scheme with the shortest path pair algorithm [16] is utilized in our simulation as the routing protocol for data/signaling forwarding. The two-fiber bidirectional mesh topology consists of more than 20 OXCs/OADMs (Optical Add/Drop Multiplexers), and we assume that the band of each fiber link is large enough to accommodate all of the path establishment requests. More than two hundred experiments were performed on our simulated path-protected THSR WDM mesh network, from Nov. 14, 2007 to Dec. 1, 2007, to evaluate our fault diagnosis performance. Among these simulation experiments, different locations and recovery times of a single fiber cut with a variety of SRLGs and dynamic light sessions are used. For each of the experiment trials, the following four methods with different combination of diagnosis information are applied, and Fig. 7 to Fig. 10 show their overall performance:

- Method(1): Consider trimmed major alarm propagation pattern only,
- Method(2): Consider Method(1) + leader major alarm domain concept,
- Method(3): Consider Methods (1) and (2) + SRLG information, and
- Method(4): Consider Methods (1), (2), and (3) + dynamic (normal) session information.

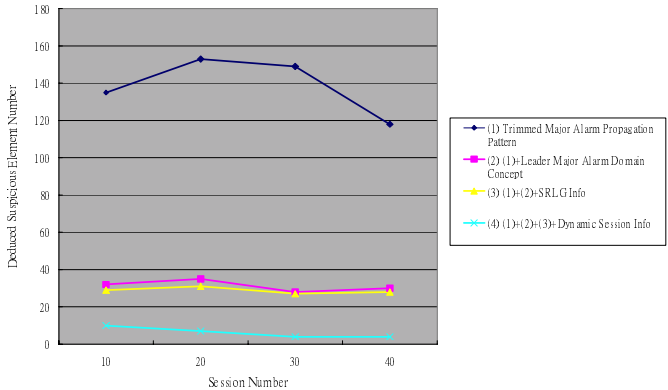


Fig. 7. Dynamic Session Number vs. Suspicious Elements

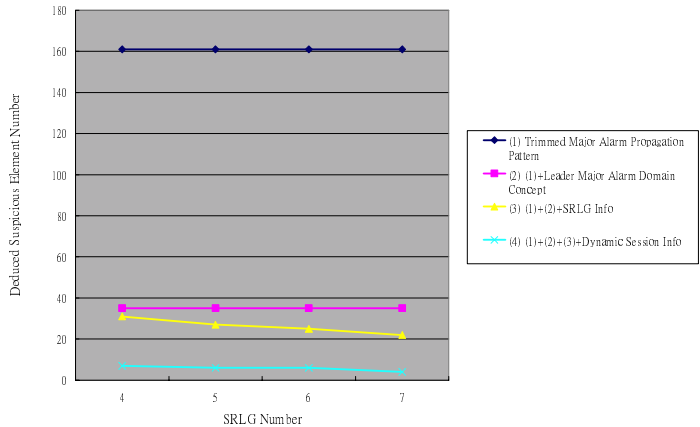


Fig. 8. SRLG Number vs. Suspicious Elements

Fig. 7 shows the average number in the suspicious element set induced by four different methods, respectively, with various dynamic light sessions and SRLGs in the simulated network. It reveals that:

- (1) Our diagnosis approach (Method (4)) using the leader major alarm domain concept and healthy diagnosis information is the most effective scheme to downsize the suspicious area for fault isolation and inspection.
- (2) The leader major alarm domain concept, dynamic session information, and SRLG information have their own “contributions,” in a decreasing order, to affect diagnosis results; using the leader major alarm domain concept, we can greatly deduce the size of the suspicious element set and with the application of SRLG, the size can be shrunk even further.
- (3) The greater the dynamic session number, the larger the probability of which more sessions remain working while faults occur. This will be followed by better

diagnosis performance with our approach since the more workable dynamic sessions remain normal/active during the fault, the more evidential information can be obtained.

On the basis of identical experiments, Fig. 8 shows interactions between the deduced suspicious element number and SRLGs, where the dynamic session number is fixed on twenty. It turns out that our approach (Method (4)) can indeed provide better diagnosis results as the SRLG number increases, even if it appears that the SRLG information has less effect on our diagnosis than the dynamic session number or the leader major alarm domain concept has.

To further examine the feasibility of our approach, effects of both length and duration (on average) of light sessions on our diagnostic approach are explored. Fig. 9 and 10 show the corresponding fault diagnostic performance to these two factors under the circumstance of which twenty light sessions are established and running on our simulated network. It can be seen that, in Fig. 9, our approach can have better diagnostic performance as the average session duration increases. It implies that the longer the light session duration is, the more information it could contribute to fault diagnoses. The same holds true in the case of session length; in Fig. 10, we can find the longer can the light session traverse, the more useful information it can collect and hence better diagnoses can be achieved for our approach. In contrast, according to results shown in Fig. 9, for networks with short light session duration, such as OBS (Optical Burst Switching) or OPS (Optical Packet Switching) networks, our approach could not perform that well.

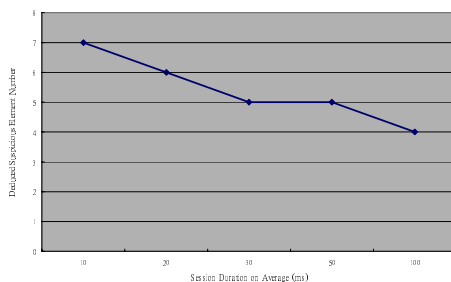


Fig. 9. Light Session (Temporal) Duration vs. Suspicious Elements

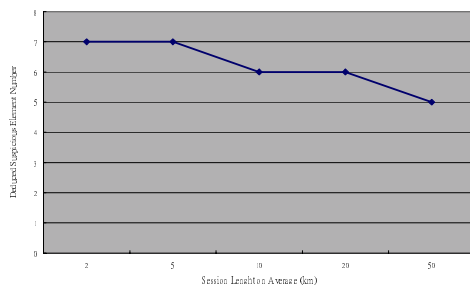


Fig. 10. Light Session (Path) Length vs. Suspicious Elements

5 Future Work

Currently, portion of the research results which includes our diagnosis approach and associated concepts have been applied and integrated to the backbone network fault management platform of the THSR, Taiwan. Although it turns out that our approach has good performance on fault isolation on path-protected WDM mesh networks, there are numerous tasks needed to be done as the improvement of our approach:

- (1) In the near future, multi-fault situations will be seriously considered in our system development due to the high possibility of the occurrence of multiple faults for some natural disasters in many countries, e.g., earthquake.
- (2) Diagnoses for various faults in path-protected WDM mesh networks will be added into the next-stage development of our approach/system to make it more practical. Plus, misdiagnosis due to alarm loss caused by the control plane network's problems would also be reviewed [17].

References

1. Perros, H.G.: Connection-Oriented Networks: SONET/SDH, ATM, MPLS, and Optical Networks, 1st edn. Wiley, Chichester (2005)
2. Zhou, D., Subramaniam, S.: Survivability in Optical Networks. *IEEE Network*, 16–23 (November/December 2000)
3. Ramamuthy, S., Mukherjee, B.: Survivable WDM Mesh Networks: Part I – Protection. In: *Proc. IEEE INFOCOM 1999*, New York, pp. 744–751 (March 1999)
4. Ho, P.H., Mouftah, H.T.: Shared Protection in Mesh WDM Networks. *IEEE Communications* 42(1), 70–76 (2004)
5. Mas, C., Thiran, P.: An Efficient Algorithm for Locating Soft and Hard Failures in WDM Networks. *IEEE Journal on Selected Areas in Communications* 18(10), 1900–1911 (2000)
6. Mas, C., Tomkos, I., Tonguz, O.K.: Failure Location Algorithm for Transparent Optical Networks. *IEEE Journal on Selected Areas in Communications* 23(8) (August 2005)
7. Sahasrabudde, L., Ramamurthy, S., Mukherjee, B.: Fault Management in IP-over-WDM Networks: WDM Protection vs. IP Restoration. *IEEE Journal on Selected Areas in Communications* 20(1), 21–33 (2002)
8. Ho, P.H., Tapolcai, J., Mouftah, H.T.: On Achieving Optimal Survivable Routing for Shared Protection in Survivable Next-Generation Internet. *IEEE Trans. on Reliability* 53(2) (June 2004)
9. Zhu, K., Mukherjee, B.: On-Line Approaches for Provisioning Connections of Different Bandwidth Granularities in WDM Mesh Networks. *IEEE OFC*, 549–551 (2002)
10. Brodie, M., Rish, I., Ma, S.: Intelligent Probing: A Cost-Effective Approach to Fault Diagnosis in Computer Networks. *IBM System Journal* 41(3), 372–385 (2002)
11. Zhang, J., Mukherjee, B.: A Review of Fault Management in WDM Mesh Networks: Basic Concepts and Research Challenges. *IEEE Networks*, 41–48 (March/April 2004)
12. Link Management Protocol – IETF Draft (draft-ietf-ccamp-lmp-03.txt) (2007)
13. Steinder, M., Sethi, A.S.: A Survey of Fault Localization Techniques in Computer Networks. *Science of Computer Programming* 53(2), 165–194 (2004)

14. Chao, C.S., Yang, D.L., Liu, A.C.: A Time-Aware Fault Diagnosis System in LAN. In: Proceedings of the 2001 IFIP/IEEE International Symposium on Integrated Network Management (IM 2001), Seattle, USA, May 14-18, pp. 499–512 (2001)
15. Katzela, I., Schwartz, M.: Schemes for Fault Identification in Communication Networks. *IEEE/ACM Transactions on Networking* 3(6), 753–764 (1995)
16. Suurballe, J.W., Tarjan, R.E.: A Quick Method for Finding Shortest Pairs of Disjoint Paths. *Networks* 14, 325–336 (1984)
17. Zuckerman, D.N.: Keynote Speech: Optical Control Plane – Management Included. In: The 10th Asia-Pacific Network Operations and Management Symposium (APNOMS 2007), Sapporo, Japan (November 2007)

Active Diagnosis of High-Level Faults in Distributed Internet Services^{*}

Huihu Long, Lu Cheng, Yongguo Zeng, and Li Wu

State Key Laboratory of Networking and Switching Technology
Beijing University of Posts and Telecommunications, China, 100876
longhuihu@hotmail.com, chenglu@bupt.edu.cn

Abstract. For fault Diagnosis in internet service, the detection and localization of high-level failure is very important and a real big challenge. The diagnose methods that passively collect information have two drawbacks: 1) requiring the target system to report its inner message; 2) it's impossible to detect and locate faults before user senses them. This paper proposes an active diagnose method which test internet service with probes and make fault inferences based on the probe results. Probing method is proactive and adaptive with low cost. We evaluate it through applying it to a J2EE application "Pet Store", compare it with a current passive method Pinpoint, and show that our method outperforms Pinpoint.

Keywords: Fault Diagnosis, High-level Fault, Dependency Matrix, Extended Dependency Matrix, Probe.

1 Introduction

With the rapid development of Internet and computer technology, internet services are becoming increasingly popular, expanding to new domains and playing an extremely important role in industry, society and human life. It is critical to make sure that internet services are running correctly and could recover from faulty state as soon as possible. Today's internet services are usually large distributed system, which makes this task even more challenging than ever before.

Internet services are usually distributed system, facing failures from different layers, say, the breakdown of network or errors of operating system. A study shows that internet services fail mostly because of configuration error and web layer logic error [3]. These faults usually change the functionality semantics of internet services, for example, a valid user can't login or a user can't add an item to his shopping cart, but not to make services inaccessible. We call such kind of faults that cause semantic errors high-level faults or application-level faults. High-level faults are the most common faults in internet service and have gained most attention from services

^{*} This paper is supported by 973 Project of China (2007CB310703), 863 Project of China (2008AA01Z201), National Natural Science Foundation of China (90604020, 90604021), Program for New Century Excellent Talents in University (NCET-07-0106), Fok Ying Tong Education Foundation (111069), and ZTE Fund

providers. Because high-level faults are usually not fail-stop faults, so some protocol-level tests, such as ping, http request, can't detect them. This paper focuses on high-level faults and we take some semantic-level method to detect and diagnose them.

1.1 Related Works

There are two representative diagnosis methods called Pinpoint and Monitor [1] [4]. Pinpoint is based on data mining approach: it traces all the requests through modifying the system message middleware and records the request id, pasted components and execution status. The recorded data will be analyzed to infer the source faulty components when faults happened. The Pinpoint have the following drawbacks: 1) it modifies the message middleware get system inner messages, which may be unfeasible or would raise the cost of deploying and maintaining the diagnosis system; 2) numerous messages arisen during monitoring will put heavy burden to the diagnosis system itself while most of the messages are useless; 3) it can only passively detect faults after they have been observed.

Similarly, Monitor needs to trace and process all the inner messages of internet services. Additionally, in order to improve accuracy, Monitor maintains a logic clock and a state graph for each component [4]. The logic clock is used to record the logic order of component calls during message processing, then a causal graph can be draw for each message. When a fault is detected, the current causal graph will be analysed to inference the faulty component. Monitor is very complicated and hard to implement. Monitor is also unable to detect and diagnose multiple concurrent faults.

1.2 Work in This Paper

This paper propose an active diagnosis method which send a customized set of probes to the managed system and make probabilistic reasoning based on the probe results. A probe is an end-to-end test, such as a ping in the network [5]. In internet services, a probe is a web interaction, a complete cycle of communication between the client and the application, and the usual situation is that a user click a button on the page and the server processed this event and return the result. For example, a user adds an item into his shopping cart. Every executed probe has a result, fail or success, and a probe succeeds only when the server returns the anticipated result.

This paper made two assumptions: (1) an internet service system is comprised of components and the components are faulty independently; (2) probe results are independent from each other. For example, the login of user A is independent from the login action of user B. These two assumptions are usually valid in current internet services. It is worth noting that although some probes may not be able to execute independently, their execution results are independent. For example, a user can't logout without login first, but the outcome of logout is independent of login.

Generally, fault diagnosis is to identify the faulty components of the managed system. In internet services, the components are mostly software modules. In contrast to network devices, the software component have two kinds of faults, the one is called one-component fault [1], which will make every call that use the component fail, and the other is called multi-component fault [1], which can only be seen under special context of components interaction. Apparently the second kind of fault is harder to

detect, and it is more common in reality because the one-component fault would usually be detected during software testing.

The main contributions of this paper are:

1) Unlike Pinpoint and Monitor, we do not need to trace inner messages of the internet services, the fault diagnosis system will be much easier to deploy and maintain. Moreover, the active probing method could detect faults before they are exposed to users; 2) A new dependency model is proposed which enables our method to diagnose multi-component faults; 3) We design a new probabilistic reasoning algorithm and use it to localize the faulty components based on probing results; 4) We evaluate our method by applying it to a J2EE application Pet store and compare it with Pinpoint. The comparison shows that our method is a little inferior to Pinpoint in diagnosing one-component fault, but performs better in diagnosing multi-component fault. This is very meaningful for the simple fact that multi-component fault is more common to see in a deployed online internet service.

The rest of this paper is organized as follows. Section2 introduces the concerned conception and the framework of active diagnosis. Section3 demonstrates this approach through applying it to a J2EE application Pet Store and evaluate the experiment results. Section4 is the conclusion and future work.

2 Active Diagnosis Algorithm

Generally, the current internet service application is a multi-tier system, usually including web, middleware and database, and each tier is comprised of a set of components like EJB, Servlet.... These components provide limited interfaces and could be treated as black boxes. A user request will trigger a sequence of component calls, and if one of the calls failed the user will get an unanticipated result which may be an http error or not. The purpose of fault diagnosis is to find the faulty component.

2.1 Dependency Matrix

As described before, an internet service system is comprised of a set of components, then a set $N = (N_1, N_2 \dots)$ could denote the system with every N_i as a concerned component [5]. Every component has a status OK or faulty respectively denoted by value 0 or 1. So $N_i=0$ means component N_i is OK, and $N_i=1$ means component N_i is faulty. A vector value $(n_1, n_2, n_3 \dots)$ will describe the system status and there are 2^n possible system statuses. In an IP network, we can think of the system as a set of devices, like router, switcher and host. In an internet service system, the components are EJB, Servlet and other class or class set. We assume that the components are independent from each other when talking about their status.

A probe is a test to the system and will pass some components [5]. In an IP network, a probe, like a ping, will pass several devices, and in internet service a probe will trigger a sequence of components call. We use $P = (P_1, P_2, P_3 \dots)$ to denote all the possible probes. Every probe will get an outcome failure or success after execution, respectively denoted by value 1 and 0. So $P_i=1$ means probe P_i failed and vice versa. Similarly we assume the probe outcomes are independent from each other, which is valid in internet service.

Table 1. A Dependency Matrix sample

	N1	N2	N3	N4	N5
P1	1	0	1	0	0
P2	1	0	1	0	0
P3	1	0	1	1	0
P4	1	1	1	0	1

Every probe will pass a sequence of components. In internet service, an interaction will make a sequence of components called. Assuming a system is comprised of 5 components: N1~N5, and we have got 4 probes: P1~P4. Then the system could be modeled by a matrix like table 1. The 1 at row i , column j means probe P_i passes component N_j and 0 means probe P_i does not pass component P_j . A column is a component and the column vector value tells which probes pass this component. A row is a probe and the row vector value tells which components the probe passes.

In a network without noisy, if a ping failed, then we believe some concerned link is down or the host is down, and if a ping success, we could believe the host and all the passed links are OK. In internet service, if a probe failed, we also believe at least one of the passed components is faulty, but when a probe succeeds, we can not predict that every passed component is OK, because a component may have a bug or error which can only be exposed in a special context. Then we can't get information from success probe, which make accurate diagnosis hard.

2.2 Extended Dependency Matrix

In order to resolve this problem, we propose the extended dependency matrix [5]. In this new matrix, a column is not a component but a possible fault. A probe pass a fault means the fault, if exist, will make the probe fail. A component may have such a fault that will make every probe that passes it fail, and we say this kind of fault on-component fault [1]. In the extend dependency matrix, we will use F_i denote the respective one-component fault of N_i . If only such faults are considered, the extended dependency matrix value will be the same as the dependency matrix. But some faults are component interaction faults and just make calls fail in special context. For example, component B has a bug and the call will fail just when component A calls component B. Then we use F_{ab} to denote such two-component fault. If a probe will

Table 2. Extended Dependency Matrix sample

	F1	F2	F3	F4	F5	F52	F41	F51
P1	1	1	1	0	0	0	0	0
P2	1	0	1	0	0	0	0	0
P3	1	1	1	1	0	1	1	1
P4	0	0	1	0	1	1	0	1

trigger such a call, then we say the probe passes the fault. Here we assume component A will always call component B in the same manner. If component A will call component B in different manners then we could model these calls as Fab1, Fab2 etc. In this way we could model three-component fault or multi-component fault.

A probe that pass component A and component B must pass Fa and Fb, but may not pass fault Fab. The matrix in table 2 presents the extended dependency matrix from the dependency matrix in table 1. The component N5 may call N2 in some context, and P4 will trigger such a call.

In the extended dependency matrix, every column is a possible fault. If it is an actual one, then every probe pass it will fail, so reversely if a probe succeeded then every fault it passed did not actually happen.

2.3 Probe Selection

Because we focus on high-level faults, so we get probes covering all the possible user actions. Put all the probes into the extended dependency matrix and we get a complete model.

In internet service, a probe can not be executed independently, for example, a user can not log out before log in. In order to execute all the probes, we must design a set of transactions to cover all the probes. It is possible that several probes could be covered just by one large transaction but also can be covered by two small transactions. In such situation we should choose that two transactions. In other words, we should choose small transactions. Assume a transaction includes probe a, b and c, and if probe a failed then we can't get the outcome of b and c. But if we use two transactions, one including a, b and the other including b, c, then it is possible for us to get the outcome of b and c.

2.4 Probabilistic Reasoning

When we get a set of probe outcomes, we use a two-phase inference to get the diagnosis result. The first phase makes a deterministic inference to get all the possible faults, and the second phase use a probabilistic reasoning algorithm to get most possible faults and the relevant components.

As described at section 2.2, if a probe succeed then every fault it passes does not happen, and if a probe failed then all the faults it passes may happen. The first phase filter faults by reserving the faults that only passed by some failed probes. The algorithm is like the following code. The inputs of the algorithm are the extended dependency matrix and the executed probes PS. The for-loop(line 2~line 8) will check every executed probe in PS, if it failed, then all the passed faults will be put in set Sf (line4), otherwise the passed fault will be put into set Ss (line 6). After the for-loop, the outcome will be Sf-SS (line 9).

```

GetSuspectFaultSet{
Input: extended dependency matrix; PS-probe Outcome
output: set of possible faults
1      S={}; Sf={}; Ss={};
2      for(p in PS)
3          if(p failed)

```



```

4         Sf=Sf+{All failure p pasted};
5     else
6         Ss=Ss+ {All failure p pasted};
7     endif
8 endfor
9 S=Sf-Ss;
10    return S;
}

```

The second phase of inference is to reorder all the possible faults from the first phase. The algorithm selects in turn the fault passed by most failed probes [2]. The code was list as follows. The inputs of the algorithm include the extended dependency matrix, the failed probe set PS and the possible fault set. The while loop (line 2~line 8) select the fault passed by most probes in PS currently. In order to do this, the algorithm maintains a map data structure to record the fault-probe count map. This data structure will be reset before every while loop (line 2).The inner loop (line 2-10) will check every probe in PS to upgrade the map data structure(line 5) . After the for loop, we get select the fault (line 7) and put it into Sf (line 9). Then the failed probes that could be explained by this fault are removed from PS (line 11). The algorithm will end when the PS is empty.

```

FaultReasoning{
Input: extended dependency matrix; PS-set of
failed probe, S-a set of possible faults
output: a sequence of faults
1     Sf={};
2     while(PS not empty)
3         ResetFaultProbeMap();
4         for(p in PS)
5             SetFaultProbeMap(p);
6         endfor
7         nf= GetMostProbableFault();
8         PS=PS-GetExplainedProbe(nf);
9         Sf=Sf+{nf};
10        endwhile
11        return Sf;
}

```

The order of the outputted faults is significant, for the front faults are more possible. We could only take K of the outputted faults to control the precision of the algorithm. The outputted faults are converted to components at the end, with one-component fault to the respective component and multi-component fault to concerned components.

3 Experiment

3.1 Experiment Setup

In order to verify our method, we apply it to a J2EE application Pet store and compare it with Pinpoint [1] [15]. The application version is Pet Store 1.4 of SUN, a three-tier

J2EE e-commerce system. We use SUN Application Server Platform Edition 8.2 as container of EJB and web component like servlets, and Derby as database. All the components are deployed on a Linux server.

As Pinpoint, we inject faults to the application, including one-component fault and two-component fault. We inject four types of faults: declared exception, undeclared exception, endless call and null call. The difference is that we also inject two concurrent faults.

Extended Dependency Matrix

As in Pinpoint, we select 52 components, including servlets and EJBs, as our diagnosis targets [1]. And we define a probe as a web interaction, which is usually triggered by user click on a button or hyperlink. We could find all the possible interactions that may be executed by a user. For example, CreateNewAccount and AddtoCart are two of them, the first one is to create a new user account and the second is to add a pet to the shopping cart.

The following matrix is a sub-matrix of our dependency matrix. For convenience, we take short name N_i to denote a component. And the mapping is: N_1 denotes ShoppingClientFacadeLocalEJB, N_2 denotes AsyncSenderEJB, N_3 denotes ShoppingCartLocalEJB, N_4 denotes CartEJBAction and N_5 denotes OrderEJBAction.

Table 3. Dependency Matrix

	N1	N2	N3	N4	N5
Changelocale	1	0	1	0	0
Sign out	1	0	1	0	0
AddtoCart	1	0	1	1	0
Submit	1	1	1	0	1

Assuming every component may have one-component fault and taking all the component interactions into account, we could generate the extended dependency matrix. The following matrix in table 4 is an extended dependency matrix, a sub matrix of the real extended dependency matrix. The first column $N_5 \& N_2$ shows that component OrderEJBAction may call AsyncSenderEJB in some context, and the probe AddtoCart or Submit will trigger this call. We only model two-component faults in this experiment.

Table 4. Extended Dependency Matrix

	N1	N2	N3	N4	N5	$N_5 \& N_2$	$N_4 \& N_1$	$N_5 \& N_1$
Changelocale	1	0	1	0	0	0	0	0
Sign out	1	0	1	0	0	0	0	0
AddtoCart	1	0	1	1	0	1	1	1
Submit	1	1	1	0	1	1	0	1

Probe Execution

As described in section 2.3, some probes can not execute independently, and we must design some transactions to cover all the probes. The transaction design strategy follows the small transaction first strategy. The table 5 lists several transactions.

Table 5. Transaction Probe Mapping

Transaction	Probe(Web-Interaction)			
Changelocale	Welcome page	Changelocale		
Sign in	Welcome page	Sign in	SignIn	
Sign out	SignOn page	SignIn	Sign out	

We use an open source tool Jameleon to automate probe execution [7]. Jameleon is an automated testing framework that manages test execution. It can detect semantic-level fault. During the transaction execution, if a probe fails, the execution will stop and the result will show which probe fails. Then the probes before this failed one in the transaction are successfully executed, and the probes after this failed one are still unknown. We will simply ignore unknown probe in diagnosis procedure.

3.2 Result Evaluation

During evaluation, we need to compare our diagnosis result with the answer: the real faulty components. For one-component fault, the answer is the respective component, and for two-component fault, the answer is the two concerned components.

We use the same metrics, accuracy and precision, as Pinpoint to evaluate our diagnose results [1]. A diagnose result is accurate only when it cover all the real faulty components. For example, the real faulty components are A, B, and the result is A or B, then neither is accurate. When we measure accuracy, we measure how often our result is accurate. Precision is the ratio between the correctly indentified faulty components and the predicted ones. If the diagnosis result is (A, B, C, D), and the answer is (A, D), then the precision is 50%.

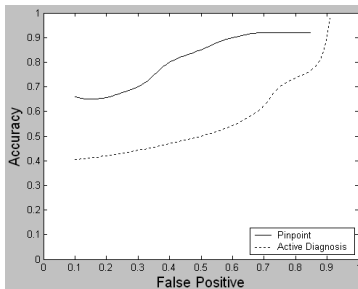


Fig. 1. One-Component fault

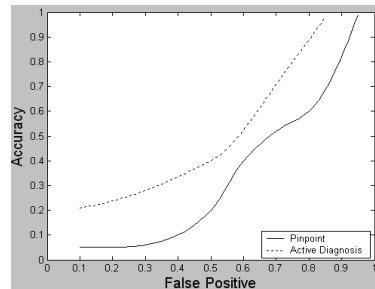


Fig. 2. Two-Component fault

As Pinpoint, in order to visualize the trade-offs between the accuracy and precision, we plot a accuracy versus false positives (1 - precision) graph when to adjust the algorithm sensitivity. In section 2.4 we show how to adjust the sensitivity by varying the result size limitation of the result set. Low sensitivity means we put more components into the result, and it will improve accuracy but lower precision.

Under the condition of one-component fault, Fig1 shows that the accuracy of Active Diagnosis is inferior to Pinpoint by 20%~30% when false positive is under 70%. When false positive is relaxed to 80%~90%, the accuracy of active diagnosis will arise greatly. When the fault is a two-component fault, the accuracy of active diagnosis will be always better than Pinpoint by 10%~20%. Active diagnosis performs well under two-component situation because of the usage of extended dependency matrix. For a probe that passes a two-component fault in the matrix, it must also pass the corresponding two one-component faults. When such a probe fails, any of these faults confirmed will make the right faulty component indentified.

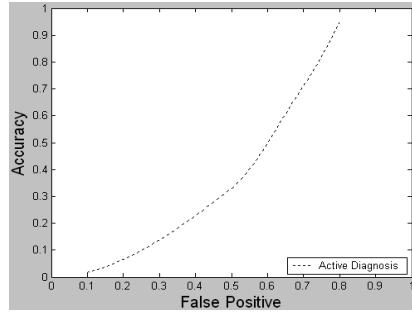


Fig. 3. Two concurrent faults

Fig 3 shows the result under two concurrent faults. Apparently multi-fault situation is hard to deal with, especially when the faults include two-component faults. We could see active diagnosis achieve 50% accuracy when false positive relax to 60%~70%.

4 Conclusion and Future Work

High-level fault diagnose is the focus of fault diagnosis in internet service. Pinpoint and Monitor passively trace all the messages of internet service during service running and make inference based on the recorded information when some fault is detected. Such kind of diagnosis methods need to modify some of the internet service module, and this make the diagnosis system hard to deploy and maintain.

Active diagnosis use customized probes to test the internet service and make probabilistic reasoning on the probing result. The active diagnosis does not need the internet service system to do anything, so it is very easy to deploy and low-cost to maintain. The experiment result shows that, although active diagnosis is a little inferior to Pinpoint under on-component fault, it perform better in diagnosing two-component faults.

The active diagnosis share some limitations with Pinpoint and Monitor. One is that we can not distinguish between tightly coupled components, for such components are always called together [1]. The other one is that we can't identify some fault that will make some request that don't pass it fail. Such situation may happen when this faulty component writes some wrong data into database, and a later request that reads the data will fail. We are trying to model this kind of fault to resolve it.

References

1. Chen, M.Y., Kıcıman, E., Fratkin, E., Fox, A., Brewer, E.: Pinpoint: problem determination in large, dynamic Internet services. In: Intl. Conf. on Dependable Systems and Networks (DSN), pp. 595–604 (2002)
2. Kompella, R.R., Yates, J., Greenberg, A., Snoeren, A.C.: Detection and Localization of Network Black Holes. In: IEEE INFOCOM (2007)
3. Oppenheimer, D., Ganapathi, A., Patterson, D.A.: why do internet services fail and what can be done about it. In: Proceedings of USITS 2003: 4th USENIX Symposium on Internet technologies and Systems, Seattle, WA, USA, March 26–28 (2003)
4. Khanna, G., Laguna, I., Arshad, F.A., Bagchi, S.: Distributed Diagnosis of Failures in a Three Tier E-Commerce System. In: 26th IEEE International Symposium on Reliable Distributed Systems
5. Rish, I., Brodie, M., Ma, S., Odintsova, N., Beygelzimer, A., Grabarnik, G., Hernandez, K.: Adaptive diagnosis in distributed systems. *IEEE Transactions on neural networks* 16(5) (September 2005)
6. Jamesleon.: <http://sourceforge.net/>
7. Brodie, M., Rish, I., Ma, S.: Intelligent probing: A cost-effective approach to fault diagnosis in computer networks. *IBM Systems Journal* 41(3) (2002)
8. Oppenheimer, D., Patterson, D.A.: Architecture operation and dependability of large-scale Internet services. *IEEE Internet Computing* (2002)
9. Yemini, A., Kliger, S.: High Speed and Robust Event Correlation. *IEEE Communication Magazine* 34(5), 82–90 (1996)
10. Lee, Iyer, R.: Software dependability in the Tandem GUARDIAN system. *IEEE Transactions on Software Engineering* 21(5) (1995)
11. Brown, I.A., Patterson, D.A.: Embracing Failure: A Case for Recovery-Oriented Computing (ROC). In: 2001 High Performance Transaction Processing Symp., Asilomar, CA (October 2001)
12. Aguilera, M.K., Mogul, J.C., Wiener, J.L., Reynolds, P., Muthitacharoen, A.: Performance debugging for distributed systems of black boxes. In: Proc. of the 19th ACM SOSP, pp. 74–89 (2003)
13. Candea, G., Kıcıman, E., Kawamoto, S., Fox, A.: Autonomous Recovery in Componentized Internet Applications. *Cluster Computing Journal* 9(1) (February 2006)
14. Pet Store J2EE Application, <http://java.sun.com/blueprints/code/index.html>
15. Cuppens, F., Mieke, A.: Alert correlation in a cooperative intrusion detection framework. In: Proceedings of the 2002 IEEE Symp. on Security and Privacy, May 12-15 (2002)

Fault Localization for Heterogeneous Networks Using Alarm Correlation on Consolidated Inventory Database

Jinsik Kim, Young-Moon Yang, Sukji Park, Sungwoo Lee, and Byungdeok Chung

KT Network Technology Laboratory,
463-1, Jeonmin-Dong, Yuseong-gu, Daejeon, Korea
{jinsikkim, xavier, suji, sungwoo, bdchung}@kt.com

Abstract. Over the decades, various fault localization techniques are introduced for the surveillance of the modern complex communication networks. But most of them analyze the events with limited data of network equipment configuration and the topology among them. But the analysis of alarm correlation based on the detailed end-to-end network view point is necessary to improve the effect of fault localization technique on complicated telecommunication networks. For this purpose, the database which supports the alarm correlation should be the integrated one which contains the information of the network elements and their connections in detail. We propose fault localization method which covers complex networks, e.g., SDH networks, IP backbone networks, IP access domain networks, xDSL networks and etc. The technique utilizes the CID(Consolidated Inventory Database) and applies the root cause event extraction method instead of problem-symptom model. And the implementations of the proposed system will be described. Then we discuss the various applications of the fault localization results on CID.

Keywords: Fault Localization, Alarm Correlation, Heterogeneous Networks, Consolidated Inventory Database, Event Normalization, Rule Base, Impact Analysis, Trouble Ticket.

1 Introduction

Most common and continuous demands for the network management society are integration and automation for the management. To reduce OPEX(Operation EXpenditure) and to survive the competitive environment of telecommunication service business, alarm surveillance and the malfunction recovery should be processed without the squandering of the human resources. Moreover IT system which is for the above purpose has a demand to be integrated one because of the maintenance cost of the IT system itself. But it tends to build the separate maintenance system for the newly-born network domains because of the distinct characteristics of the new network, even though the telecommunication company is under the strict policy and guidance for the system integration. So it is hard to provide network surveillance of entire network view point with these divided monitoring

systems. For instance, IPTV service is provided under the combination of various legacy networks. So if there are faults on the IPTV service, the related surveillance systems should be checked through the complicated communication among the human operators of each own network domain.

Because of the above inconvenience we propose the fault localization method using integrated network alarm correlation technique based on CID which stores the network equipments details and the connection data among them[1]. The proposed method collects the network alarms from various NMSes(Network Management System) which manages its own network domain. And then it analyzes the collected events with rule-based alarm correlation method using CID and predefined correlation rules. It uses RCEE(Root Cause Event Extraction) method instead of problem-symptom model[2] which has been widely preferred way for alarm correlation. RCEE method extracts the representative event, so called root cause event, among the grouped events which are assumed to be related with each other by the correlations rules. If the extracted event can't represent the real fault, we propose to generate the presumptive event. So it prevents the system from losing the advantage of the problem-symptom model which points the exact inferential problem which caused the symptoms. And we propose the other applications of fault localization results on CID. For example, the damaged lines and customers can be estimated in real time just after finding the root cause event.

The section 2 introduces the entire system of fault localization based on CID and explains CID which is the fundamental integrated facility and link database. And it shows the key processes of fault localization for the heterogeneous networks. The detail comparison between RCEE and the problem-symptom method will be done. The section 3 proposes the applications of fault localization results on CID. The section 4 illustrates the implementation of proposed system. And then we will conclude that and discuss the further researches in the section 5.

2 System Representation

Firstly, we will show you the entire of the proposed system in a brief way. Then we will explain each module in detail.

2.1 System Overview

Network alarms are collected from NMSes of each network domain or alarm management servers. The format of each domain events are already organized to be processed for each own system. Therefore the events should be normalized by converting the raw event to the proper format of CID view point on event normalization step of Fig.1. On this process, the occurred location of event is changed to follow the form of CID standardized way. The surveillance organization code for the event are recorded to be filtered out for the unconcerned department when it is determined to be a root cause event. The related events are grouped by various predefined rules which are considering the occurred location and temporal relations, etc. Finally it extracts the representative root cause event from each event group on alarm correlation step. And all the processes of collecting, converting, mapping, analyzing the events are based on CID.

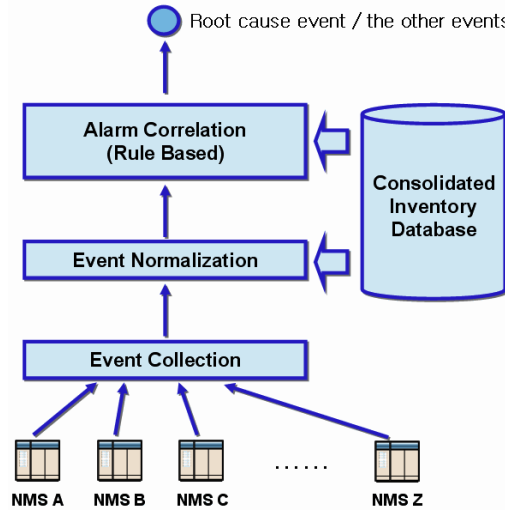


Fig. 1. Fault Localization System, Event Flow Diagram. Events are collected from various NMSes and normalized based on consolidated inventory database. On the alarm correlation step, it extracts root cause event with predefined correlation rules.

2.2 Consolidated Inventory Database

CID is the integrated facility and circuit, link, customer database[1] which is fundamental for the proposed alarm correlation analysis. On multiple domain alarm correlation, the inference process of fault localization can be limited if there is no network equipment and connection data. CID can store the network data by modeling the equipment configuration and links among them. Then it manages the stored data by synchronizing the real equipment and link status to the database constantly. In addition to network data, the common data for network management. For example management organization, equipment location on the map, customers allocated on each equipment, various cables are also stored. CID on this paper modeled the NE(network element) as the structure which is composed of rack/shelf/slot/unit/port objects. For the transport network equipment, the carrier is modeled as the circuit between two equipments and has its own unique code. Each channel line of carrier also has the similar characteristics of carrier except that it can contain another lower capacity carrier. For IP equipments, they have physical and logical port. And the connection data for each type of ports are established. To make the connection data complete with E2E(end-to-end) view point, it stored which channel of carrier transports which link between IP equipments when the IP routers are connected through transport networks. So if the carrier in transport networks has the problem, we can know that the allocated IP connection also has the damage by detecting ping down event as shown on Fig 2. Therefore, from the customer to the other customer, all the connections, ports, equipments can be fully understood base on CID. we can extract the root cause event among the multiple events from various network domains using the E2E topology of CID.

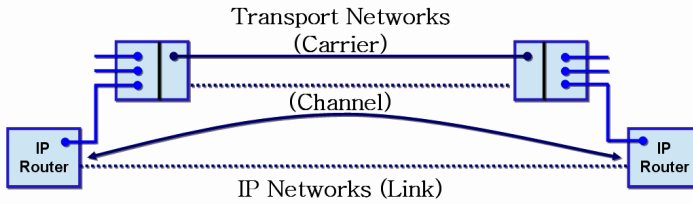


Fig. 2. Transport Networks(SDH Networks) / IP Networks relation sample, one channel of a carrier carries the data between routers. So if we have a problem on the channel, then we know that each router emits alarms. All these cross domain topology information are stored on CID.

2.3 Event Collection and Normalization

The functions of event collection and normalization processes are that it collects the NE alarms from each NMS first and then converts the raw events to specific normalized format which is composed of the original raw event data, CID mapped occurred location, alarm correlation analysis knowledge data.

Each NEs has its own format of alarm event because it is difference equipment. And Each NMS also has its own normalized event format for managing the different type of network elements. Therefore, If the whole alarms of each NMS are gathered together, it's hard to analyze the alarms on integrated network view because of the format difference. Thus in advance to fault localization process, we should transform the different events to the specific format to analyze the alarms fast and efficiently.

It's more efficient to make the NMSes to follow the standardized form of CID style equipment location before the fault localization system transforms those events. But some NMSes have existed before CID is constructed and sometimes it's hard to change the whole system. Therefore CID based equipment location and responsible organization code should be added by event normalization. For that, it should have occurred location mapping process on normalization step for each NMS. This Mapping process mainly converts the raw occurred location data to CID style location code. In case of non-IP NEs, e.g., NE of transport networks or ATM networks, the location data can be hard to be converted for its complexity. So it is necessary to construct mapping table. And it should be editable by the operator for each location and each kind of equipment. Because the mapping table can be added or modified when the networks are expanded or upgraded.

And time data should be normalized also. The fact that temporal relation is the key information to group events is widely known[3]. But exact occurred time data of events are hard to know because of the incomplete time synchronization of whole networks. Moreover each NMS determines the occurred time as their own way. For instance, they choose the event generated time of the network equipment or the time to collects the events as event occurred time. Thus the normalized occurred time is necessary to minimize the time synchronization error of each network domain. For well time-synchronized network domain, the time data aren't need to be changed. But For the domain which has irregular time-synchronized NEs, the collection time is more reliable as event occurred time. The other traditional event data are also necessary, e.g., severity, alarm kind, occurrence/clear status should be recorded on this step.

In addition to the above data, event analysis knowledge information is also the normalization target. For instance, the events are occurred on both sides of carrier on SDH network element. And if we have information of alarm priority value for each alarm kind, the root cause events can be inferred by the value comparison. The domain experts of the NE has the knowledge of the priority for each alarm kind. Thus the experts allocates the proper alarm level value for each alarm kind and so the alarm level value of the event is determined on the normalization process according to the alarm kind of the event and value mapping table. Another example is 'self-caused' or 'other-caused' parameter. On Fig.2, if the drop part of channel emits the event of other-caused event, e.g., LOS(Loss Of Signal), then the alarm of the connected router is root cause event. It is normally used to determine the priority between the cross domain connection line which are on the same layer.

Service affect tag which indicates the yes or no for that the occurred event causes service trouble or not is another analysis knowledge. Some NEs give the service affect information when the event is occurred. But for other NEs, domain experts of each network constructs the service affect mapping table for each alarm kind of each NE. And the data are used for the other applications of alarm correlation. It will be explained on section 4.

The determination that the event is caused by usual network management operations, e.g., a replacement or expansion of the equipment, can be another data. If certain NE is under construction for the upgrade, then it will emit various alarms on specific time period, the construction schedule should be referenced on alarm correlation step.

As we discuss above, the raw event data, CID mapped data, alarm analysis knowledge are normalized before the alarm correlation analysis for efficient and fast fault localization. The details can be different for the target network domains which are covered on alarm correlation and for the network management infrastructure and alarm correlation rules to use.

2.4 Alarm Correlation

There were many kinds of alarm correlation researches[4]. Most of them have focused on problem-symptom model.[1]. It constructs the problem and symptom relation model before real time analysis of the alarms, and then the problem can be inferred from observed symptoms by comparing the existed relation. Usually Model-Based approach or Codebook approach[5] uses this problem-symptom model. But on this paper, we assume that fault localization is to find representative event(root cause event) among occurred events and name it as RCEE(Root Cause Event Extraction). RCEE chooses the root cause event among the grouped events by the alarm correlation rules. And the rules are based on time sliding window and integrated network topology relation of CID. On problem-symptom model, there are too many problem-symptom relations to be constructed on modern multiple networks of multiple equipments. For instance, CID of the proposed system has 7 thousands types of NEs and if we have 10 problem-symptom relation for each NE, then we need 70 thousands of relations to be defined and it still doesn't consider the multiple NEs problem-symptom relations which are necessary to analyze the cross domain problem. But on RCEE method, there is no need to define the problem-symptom relations. It just extracts the high priority and representative event among the grouped events. Moreover, the event which is determined to be a root cause event usually represents the fault because the event is occurred on suspicious location of the NE.

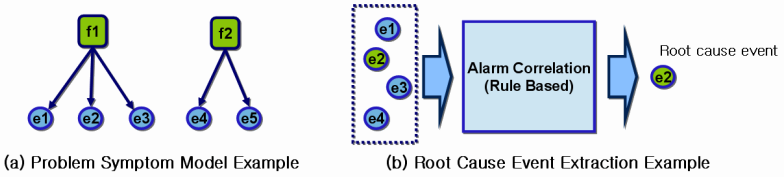


Fig. 3. Problem – symptom model try to find the fault from occurred events. And RCEE methods extracts root cause event from related event group using alarm correlation rules.

But in some cases, the chosen representative event can't be a root cause event. The most common case is that the root cause event is not occurred or missing on event collection process. For instance, if there are upper layer L3 switch and other many connected lower L2 switches on IP networks. If the alarms occurs from all the L2 switches but not from L3 switch, then it can be inferred that L3 switch has problem even though there is no alarm from L3 switch. So it is the weakness of the RCEE strategy compared to problem-symptom model because the model is designed to point the exact inferred fault location.

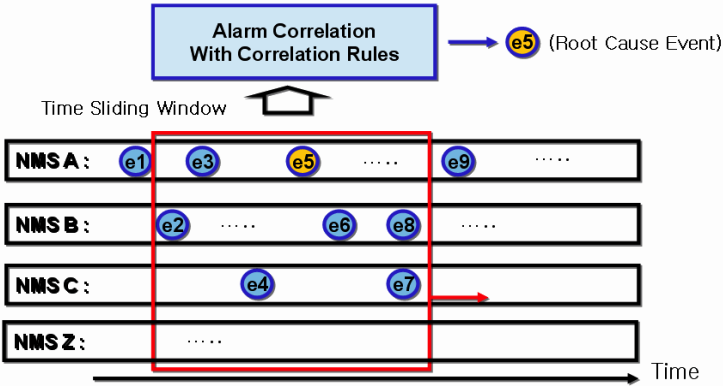


Fig. 4. Alarm Correlation Example. The events from each NMSes are grouped by the criterion of sliding time window boundary. And alarm correlation process determines the root cause event based on the predefined correlation rules. 'e5' is root cause event in the above case.

So for correcting the weakness of RCEE, we propose to generate a presumptive event on the alarm correlation analysis process for the above root cause event missing or delaying. If it is inferred that there is a missing root cause event by the alarm correlation rules, the presumptive event is generated and handled like other normal events. It means that the presumptive event also has same information with other normal events, e.g., a exact equipment code, time and location on CID data with other normalized events. The occurred location is inferred from the other related events. For the above L2, L3 cases. The L3 equipment itself is the location of the presumptive event and it becomes root cause event.

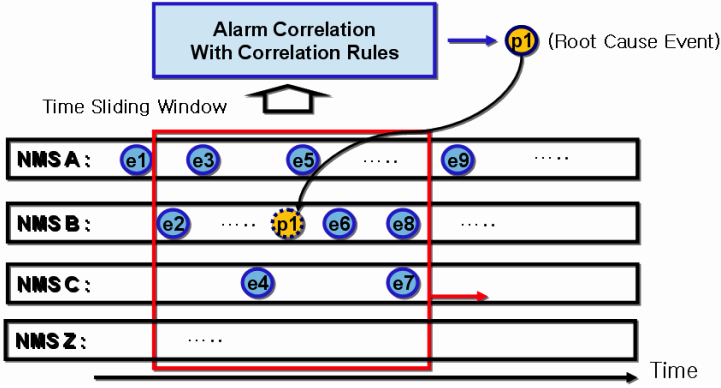


Fig. 5. Presumptive event generation example. Alarm correlation process generates the ‘p1’ presumptive event based on the correlation rules. if necessary, the correlation process generates virtual event to point exact fault location(Compared with Fig.4).

Problem-symptom model has the advantage that it can detect the exact fault even though there are some missing symptoms. RCEE and presumptive event generation strategy has the flexibility which the problem-symptom model has, and also it keeps the fault localization problem simple because there is no complex model which should be defined first. In other words the proposed method has the effect of having problem-symptom model for exceptional situation of missing event case, but it still keeps simple and cost effective RCEE strategy which covers most of fault localization cases.

Moreover we generate presumptive event for any element on CID, even if the element never emits alarms. For example, if there are large number of events than predefined limit on circuits and links carried by specific optical fiber cable, we generate presumptive event for the optical fiber cable on CID. Thus proactive fault management is possible by developing and enhancing the rules of presumptive event generation because we can know the fault before the alarm is occurred. But if there are too many generation rules, the unreliable event can bother the network operators, hence the degree of strict on generation rules should be controlled.

Alarm correlation rules of the proposed system are mainly considering temporal and topological relation of the occurred events. We have four steps of rule process. First step is preparation before event grouping like event suppression. Second step is about event grouping based on network topology and root cause extraction. Third step is mainly focus on generating presumptive event. Forth step is different from others. It is executed continuously to handle time sliding window or to clear the non-cleared events. The details of the rules can be different according to the handled network domain and CID.

3 Applications of Proposed Alarm Correlation Result

We propose the following applications of using root cause event which is based on CID.

3.1 Impact Analysis

It is possible to estimate the impact for the root cause event because the event has the normalized location on CID. If the event occurred location is a optical fiber cable, then the circuit and links which is transported by the cable are found on CID. By tracking the circuits and links to the customer side equipment, all the circuits and service customers can be listed because allocated customer IDs for all the ports of the service edge equipment are stored on CID. In the past when CID is not constructed and there is no end-to-end connection data, the network operators have to investigate and combine their own domain analysis result manually for estimating the impact of certain fault. Thus it took few days or weeks. But based on real time fault localization and the end-to-end network data on CID, the impact analysis can be automatically done just after the fault happens. In other word, After checking the service affect tag of the root cause event, damaged lines and service affected customers can be extracted by querying on CID. The service affect tag is mentioned on section 2.3.

Furthermore the impact analysis results can be utilized to provide other useful functions. If there is a process to find the customer who calls the call center on the impact analysis results in real time. Then the operator knows the root cause event which cause service malfunction to the customer directly, Because the service affected customers are already listed by the fault localization and the impact analysis before the customer knows the malfunction and usually the alarms are collected just after the fault happens. The other function is customer care service. For instance, if certain important customers are listed by the impact analysis of certain root cause event, then the member of the customer care department can know the error. Consequently he has a time to cope with the emergency before the customer try to contact the person in charge. So it can be useful to care the important customers.

3.2 Trouble Ticketing

Trouble ticketing is to track the detection, reporting, and the resolution of network problem. Fault localization technique is originally for resolving the malfunction. So the trouble ticketing base on the above root cause event is fastest way to fix the problem. Because the root cause event now has the exact location of fault, damaged customers list and, the number of customers to call and the whole data can be provided right after the fault occurs.

Each network domain has the criterion to issue the trouble ticket. The number of customer calls is usually certain criterion, and the bandwidth of the service line of the fault is also referenced. The number of impact customers, the downtime, importance of the equipment are also the useful criterion for controlling the degree of frequency level of trouble ticket issuance.

And those trouble ticket statistics can be the analysis back data to improve the alarm correlation rules. Because there are detailed records for each root cause determination on trouble ticket.

4 The Implementation

We use CID which handles approximately 3 millions of NEs which are classified into 7 thousands kinds of NEs. And it supports 30 millions of network service contracts.

The size of whole data is about 200 Gbytes including NEs, cables, physical lines, logical lines, customers, buildings and organizations. It supports xDSL(x Digital Subscriber Line), leased line, Ethernet services. Alarm correlation covers the following network domains of SDH networks, IP backbone networks, IP access networks, NAS/DSLAM for xDSL networks, ATM networks, PSTN networks, etc. Event collection and normalization, alarm correlation is implemented as the proposed method and additional features are added for real-world problem but the basic architecture is not different. For alarm correlation, the commercial rule engine is used. The implemented system has hundreds of alarm correlation rules which are collected from each domain experts group. The rules handle temporal relations, topological relations, cross-domain priorities and each domain specific decisions, etc.

Based on the above alarm correlation rules, all the events are reduced to average 10 % of root cause events. It means that 10 events are grouped together and one of the events becomes root cause event of the group. We also implement the impact analysis, trouble ticketing, customer call matching functions. In addition to the applications, the system provides scheduled network equipment/line construction related information by tagging the root cause event as network construction event and related statistics for fault and trouble tickets are also provided on the system.

The system is now on working to watch the networks on the view of integrated networks, to issue integrated network trouble tickets base on the fault localization result, and to care important customers by using the impact analysis results

5 Conclusions and Future Works

On this thesis, we propose the fault localization technique for heterogeneous networks based on consolidated inventory database(CID). For the complexity of integrated networks, root cause event extraction and presumptive event generation method are introduced. the proposed strategy keeps the fault localization problem simple but still has the advantage of flexibility which problem-symptom model shows on event missing. And we also introduced impact analysis, customer service care, trouble ticketing applications for the fault localization result on CID. Finally we prove the system can be applied to real-world network management by implementing and operating it.

But the proposed system still watches only network alarms. Recently many of researches try to cover service faults and the upper layer part of physical networks[6][7]. Thus the additional researches are needed to extend the surveillance coverage to service area for each proposed step, e.g., service related event collection and normalization, CID for the service resource elements and dependencies, alarm correlations rules for the service alarm event correlation.

References

1. Lee, K.T., Jin, E.S., Yun, D.S., Jeong, J.W.: A Replication Architecture of Central Inventory Database. In: Proceedings of the Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services, pp. 39–45 (2006)

2. Jakobson, G., Weissman, M.D.: Alarm Correlation. *IEEE Network* 7, 52–59 (1993)
3. Steinder, M., Sethi, A.S.: A survey of fault localization techniques in computer networks. *Science of Computer Programming* 53, 165–194 (2004)
4. Gardner, R.D., Harle, D.A.: Methods and systems for alarm correlation. In: *GLOBECOM* 1996, vol. 1, pp. 18–22 (1996)
5. Yemini, S.A., Kliger, S., Mozes, E., Yemini, O.D.: High Speed and robust event correlation. *IEEE Communications Magazine* 34, 82–90 (1996)
6. Hanemann, A.: A Hybrid Rule-Based/Case-Based Reasoning Approach for Service Fault Diagnosis. In: *Proceedings of the 20th International Conference on Advanced Information Networking and Application*, vol. 2, pp. 734–740 (2006)
7. Cheng, L., Qiu, X.S., Zhang, N.: A Service-Oriented Alarm Correlation Method for IT Service Fault Localization. In: *Communications and Networking in China, ChinaCom* apos, pp. 1–5 (2006)

Probing Based Topology Inference for Large Scale Community Networks

Marat Zhanikeev¹, Yoshiaki Tanaka^{2,3}, and Tomohiko Ogishi⁴

¹ School of International Liberal Studies, Waseda University
1-21-1 Nishi-Waseda, Shinjuku-ku, Tokyo, 169-0051 Japan

² Global Information and Telecommunication Institute, Waseda University
1-3-10 Nishi-Waseda, Shinjuku-ku, Tokyo, 169-0051 Japan

³ Research Institute for Science and Engineering, Waseda University
17 Kikuicho, Shinjuku-ku, Tokyo, 162-0044 Japan

⁴ KDDI R&D Laboratories

2-1-15 Ohara, Fujimino-shi, Saitama 356-8502, Japan

maratishe@aoni.waseda.jp, ytanaka@waseda.jp, ogishi@kddilabs.jp

Abstract. Traditional research in on-demand topological solutions is gathered around the two main clusters, - wireless ad-hoc networks and fixed overlay networks. Both are very different in nature, but they both deal with the same problem which is to create a topology out of an arbitrary set of nodes. This paper considers the case of an arbitrary set of mixed-technology nodes which are to be joined in a topology based on end-to-end delay measurements among nodes. The core of the proposal is topology inference based on triangular inequality of end-to-end delay which is finalized in form of an algorithm that allows for efficient detection of a logical topology of a network with no initial topology. The algorithm is scalable and could be a practical solution for many scenarios involving community services created on-demand and intended for a short lifespan.

1 Introduction

The number of global community networks is constantly growing. Skype, PlanetLab, Skitter, etc. are only a few well known names in today's global network. Each of them uses some form of network embedding, - the term used to describe translation of end-to-end delay measurements into Euclidian space.

Topology inference in large-scale community networks becomes difficult when the number of nodes in the network is large and performing $O(n^2)$ measurements is not practical. Additionally, having all-to-all delay samples does not always solve the topology inference problem which seeks to detect logical topology of a community network rather than Euclidean translation of node positions. In fact, authors of this paper are not aware of an algorithm that would be able to infer topology based on raw end-to-end measurements among all individual nodes in a community network.

Network embedding is also becoming an important network service demanded by global community networks. Vivaldi [5] and Meridian [6] are the two most

popular models used for such purpose. Vivaldi is slightly inferior to Meridian in precision which comes from the difference of approaches these models use for global-scale network embedding.

The problem with traditional network embedding is the incremental nature of the process. At any point of time in the lifespan of a community network a new node arrives to be integrated into the topology. This triggers the process of neighbor discovery which is the source of major differences between Vivaldi and Meridian. Both models use a number of reference nodes selected from already existing topology to triangulate the newly arrived member and finally positioning it at some coordinates in Euclidean space. These coordinates can be used to find a communication partner which normally would be one of the nodes located nearby. While creating topologies based on Euclidean coordinates is a trivial task, the problem is that Euclidean coordinates do not necessarily represent the logical topology on the network.

This paper proposes a simplified algorithm for topological inference in large-scale community networks without attributing Euclidean coordinates to each node. The target of this research is to reveal logical topology with as little measurement overhead as possible. Results are verified on a random physical topology generated by the GT-ITM topology generator [1].

2 Delay Properties of Community Networks

Most research in network embedding is conducted on sets of end-to-end delay measurements obtained from existing global community networks. Popular delay data sets are *p2psim* [7], *DS²* [8], and *PlanetLab*. Each data set contains delay measurements in meshes containing from several hundred to several thousand nodes. However, no connectivity graphs are provided with these data sets which means there is no way to verify how well results of network embedding match actual geographical locations of nodes and underlying physical structure of the network. This section discusses some of the issues related to delay properties and physical topologies.

2.1 Triangular Inequality Basics

Close proximity in terms of physical connectivity is the main source of error in neighbor discovery in both Vivaldi and Meridian models. In case of a triangle containing nodes A , B , and C , and $d(AB)$ signifying the delay between nodes A and B , the less the physical distance between the nodes, the more frequently $d(AC) = d(AB) + d(BC)$. Literally, node B would often be in the middle of the only possible path from A to C , i.e. A , B and C would be on a straight line.

For example, all four connectivity cases in Fig.1 would result in identical delay data sets. As a rule of thumb, the farther the nodes in neighbor discovery are from each other the more reliable is the result of triangular delay measurements. Additionally, neighbor detection with reference nodes farther apart makes more sense when the target is a logical topology of the network rather than Euclidean coordinates.

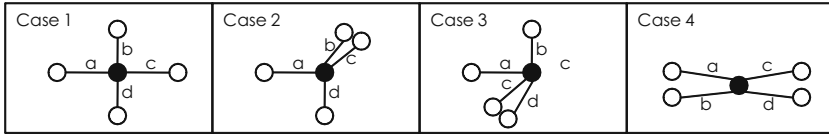


Fig. 1. Given that legs a , b , c , and d are equal in on-way-delay, all the four cases above will produce perfectly identical results

2.2 Delay Maps and Physical Topologies

Consider three three cases in Fig.2. In all three cases one-way delay measurements are conducted from two reference nodes marked $R1$ and $R2$. The reason two nodes are used for the measurement is so that a cross-reference of network-wide delay distribution can be obtained. This is also a small-scale version of the proposed algorithm that will be proposed later in the paper.

Delay maps for each of the three cases are created using unit distance from $R1$ as horizontal and unit distance from $R2$ as vertical coordinate in the chart. Each leg in all topologies is exactly one unit of delay long. All topologies contain a fork in connectivity.

The visualizations in Fig.2 should be interpreted as follows. When both reference nodes are placed on the same branch of the tree, the fork is not present in the delay map. In this case cross-references in delay measurements have no significance. Cases 2 and 3, however, are correctly reflected in the delay space. More than that, all bridges between branches purposely placed in Case 3 are translated as bridges in delay space as well. This feature is exploited in the proposed algorithm in a more sophisticated form. It should be noted that delay maps are significant only when reference nodes are placed on different branches in the tree. In the proposed algorithm this element is specifically ensured.

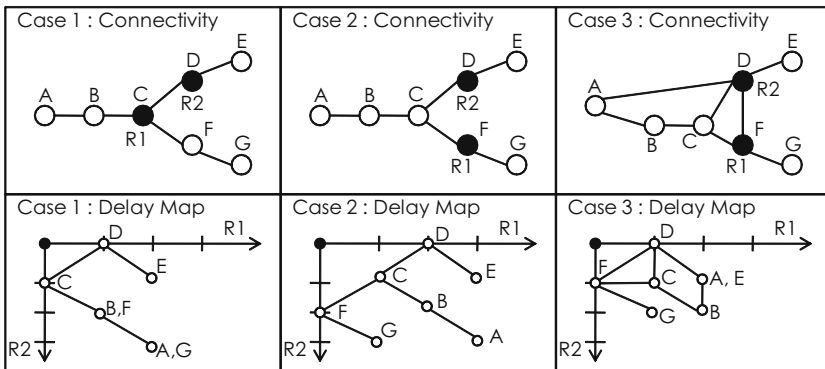


Fig. 2. Three examples of how physical connectivity can be translated into delay maps by using two reference nodes located at various points within the network

3 Topology Inference Algorithm

The fact that delay-based view of a community network is different from its geographical view offers room for topological inference. Based on delay jumps and perturbations as described in [2], it can be assumed that delay-based map of a community network should consist of a number of clusters separated by relatively long intra-AS spans. This section makes a number of such assumptions and proposes an algorithm that exploits them for the purpose of topological inference.

3.1 Topological Assumptions

In the analysis of end-to-end delays in large-scale community networks the following assumptions are in place:

- distribution of delays is mostly presented in the two extremes, - either very long or very short delays;
- many short delays form clusters which are also located nearby in geographical location;
- long delays lead to major crossroads in the community network, which is not the case with short delays;
- virtual centers of community networks are located equally far from all edges as viewed by end-to-end delay.

Apart from the above assumptions, it is also important to minimize differences in inferred topology depending on which node in a community network starts the process of inference. Contemporary community networks are normally created by an incremental process, which enables continuous expansion of the network as new members arrive. This is true for both PlanetLab and Skype, - the two largest community networks in the world today. Incremental process allows for certain freedom in topological inference, since these networks started from a relatively small number of nodes and kept growing over the years.

It can be foreseen, however, that NGN networks will bring about a new way of creating community networks by giving full capabilities to a large community of mobile users. In this new environment many community networks will have to be created on-the-fly to exist only for a short period of time. Topological inference in such community networks will be the issue of utmost importance.

Additionally, the above futuristic scenario will impose limitations on the performance of a method used to infer topologies of newly created community networks. Literally, there will be no time or processing capacity for $O(2^n)$ end-to-end measurements to define how far members of a given community network are located from each other. The number of measurements should be minimal yet offer a solid delay-based view of the network. In the best case, $O(mn)$ complexity is expected, where m is the number of times measurements from a single node to all other nodes is performed, n being the number of nodes and $m < n$. Smaller m values result in faster results of topology inference and are good especially for on-demand communities with short lifespan.

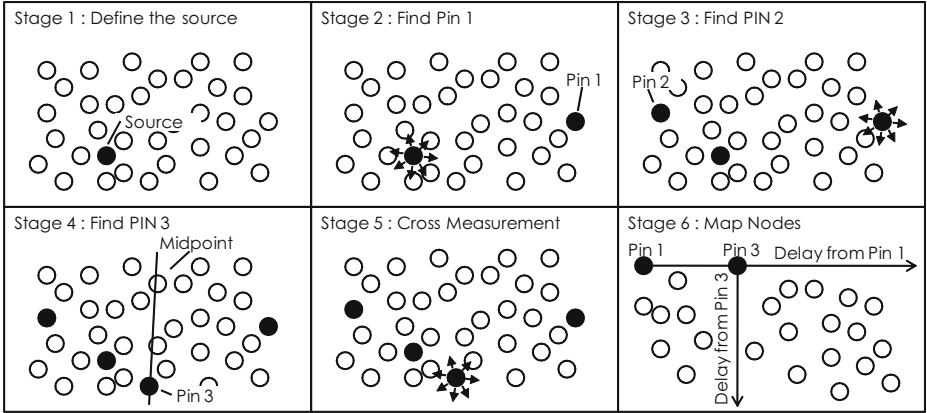


Fig. 3. Stepwise process of topology inference

3.2 The Algorithm

Fig.3 displays the overall process required to complete topological inference in an arbitrary size community network. Most steps involve end-to-end measurements from a single node to all the other nodes in the community network. Each step is performed as follows.

Stage 1: Define the source is the initial step for any community network. Although in this paper the source is selected by random selection, in real like many scenarios are possible. Such as, in content delivery networks this could be the node that has the content and needs to distribute it among community members. Since this paper selects the source randomly, it can satisfy any scenario used in practice.

Stage 2: Find Pin 1 involves end-to-end measurements originating from the source and terminated at all other nodes in the community network. Therefore, with n nodes in the network, $n - 1$ measurements are performed. The purpose of the measurement is to find end-to-end delay between the source and another node. In real networks, round-trip delay measurements can be successfully used instead of one-way delay. Since this paper uses simulated topologies, it is easier to operate with one-way delays which are calculated as the sum of delays of all individual hops on each end-to-end path. End-to-end paths, in turn, are calculated using the shortest path algorithm.

Stage 3: Find Pin 2 repeats end-to-end measurements, this time using Pin 1 as the source. The purpose of this set of measurements is to find the most distant node from the Pin 1, thus, defining the boundaries of the entire community network. At the end of Stage 2, the set of delay values represents one-dimensional view of delay distribution of the entire community network. To add the second dimension additional steps are required for further details.

Stage 4: Find Pin 3 is used to create 2-dimensional delay map of the community network. To do this, a mid-point between Pin 1 and Pin 2 is selected as node with the least difference between delays from Pin 1 and Pin 2, i.e. the selected node would be virtually in the middle between Pin 1 and Pin 2. In the result, Pin 3 is obtained and will be used to create 2D delay image of the network.

Stage 5: Cross Measurement is the fourth set of measurements originating from Pin 3. This set of delays allows for cross-reference with delays originating from Pin 1 and Pin 2.

Stage 6: Mapping is performed by using delays originated from Pin 1 and Pin 3 as coordinates in the delay-based map. Pin 1 is always zero on horizontal axis but may not be zero on the vertical. Pin 3 is the zero on the vertical axis but may be placed at a positive value on the horizontal axis. In the result of this operation, each node in the community network is attributed with 2D coordinates based on end-to-end delay.

3.3 Simulation Results

From the extensive search for a topology generator, GT-ITM appears to be the only topology generator that can create multiple-tier topologies. Its closest competitor would be the BRITE topology generator but the latter generates only flat topologies which are meaningless for delay-based topological inference performed in this paper. All the results below are based on a randomly generated multiple-tier topology with 100 nodes.

Before the results of the proposed algorithm are presented it is important to view the underlying topology. Since a randomly generated topology is used to obtain simulation results, actual delays can be analyzed based on the actual graph. Fig.4 is used to visualize logical topology of the randomly generated graph as viewed by one-way delay. The center of the delay-based view was selected arbitrarily by the process of minimizing variance of hop count of end-to-end shortest paths from the central node to all other nodes in the network while also making it closer to the average hop count among all shortest paths in the network. This way the central node is literally the closest node to the center of all delays in the community network.

An important feature of the delay map in Fig.4 is the use of actual delays as Euclidean distances in the visualization. Actual delays are marked by thicker lines and are true only as long as they stay on the same branch.

The visualization in Fig.4 is constructed as follows. After the central node is defined, the visual is constructed incrementally by adding neighbors for each node starting from the central node. Neighbors of the central node naturally form the main branches. All central branches split the entire area in sectors of equal angles. Every following branches equally split smaller angle ranges in such way that branching would continue in the outward direction from the center, thus, creating an easily digestible visual image.

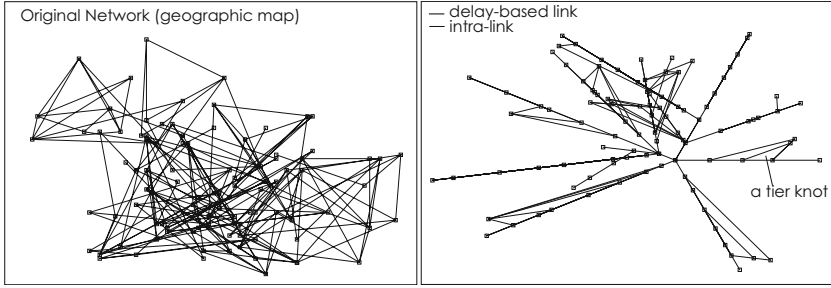


Fig. 4. Raw geometrical graph and delay-based view of the same graph

Thin lines in Fig.4 represent intra-domain connections that normally stand for meshes within a network tier or intra-AS connections. They are not plotted during the branching process but are added to the visualization at the end. However, they play an important role in understanding topological properties of multiple-tier network graphs. Judging from Fig.4, intra-links create only local irregularities, leaving the overall branching structure intact. When links are connecting nodes on the same tier, the structure could be referred to as a tier knot or a tier mesh.

Given that the visualization in Fig.4 is based on the actual graph, it will be important later when topology inference results are discussed. One major difference between the delay map in Fig.4 and topology inference results is the fact that topology inference results are based only on relative one-way distances and should not be interpreted as Euclidean translations.

Fig.5 displays results from using the proposed algorithm on the same randomly generated multiple-tier network graph. The Case 1 of the delay map offers a clear view of the network and its topological structure. The very upmost node is the Pin 3, i.e. the perfect center of the community network as viewed from both Pin 1 and Pin 2. It may not be the perfect center of the horizontal axis, however, given that the geographical distribution of nodes is irregular.

Since the delay map is created from the virtual left and the virtual center of the network all logical forks in topology are easily revealed as visual forks in the map. This has a physical explanation related to connectivity. When there is a connectivity fork in the underlying physical networks, distances on each side of the fork change independently from each other when translated by delay from the edges of the community network. This independence in change further away from a connectivity fork directly translates into a distance in the delay map. How forks in the networks translate into delay maps was already discussed earlier in this paper on a simpler case.

Topological mapping in Case 1 of Fig.5 does not detect all topological forks. In fact, the more points are used to create delay cross-references, the more forks can be detected in the resulting delay map. Cases 2 and 3 in Fig.5 are the examples of such precision boost coming from adding more reference points in the network.

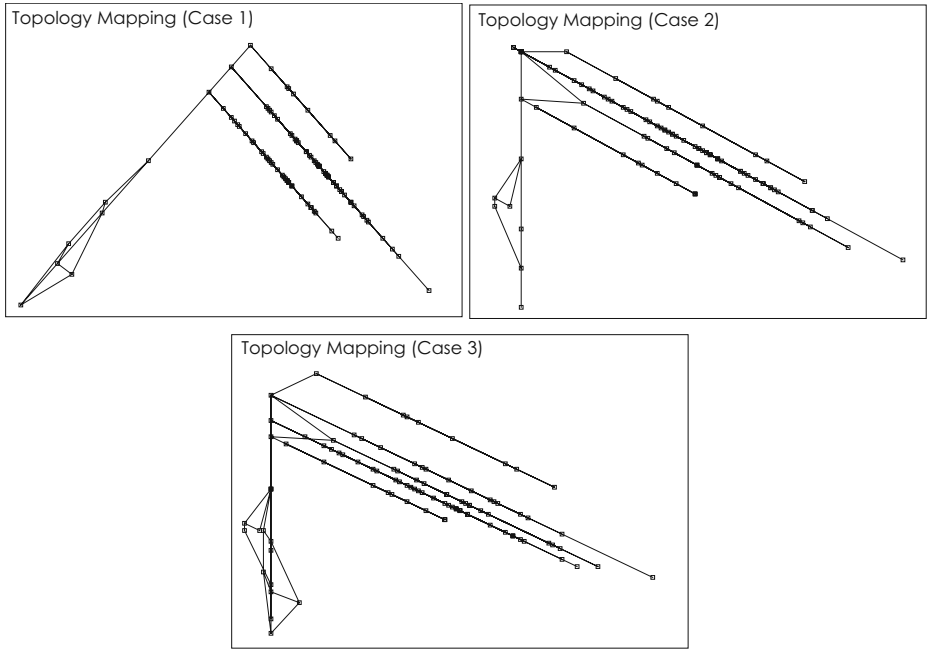


Fig. 5. Original network with geographic locations and three cases of topologies inferred from delay maps

To create delay maps in Case 2, instead of using a single node in the center of the network, two nodes at 0.33 and 0.66 of the delay distance between Pin 1 and Pin 2 were used. Accounting for geographical irregularity, the closest nodes to the above positions were selected. In the end, vertical coordinate was obtained as an average of the delay to a node from the two central points. This resulted in more forks revealed through the delay map.

Finally, Case 3 in Fig.5 uses 3 central reference points selected at equal intervals between Pin 1 and Pin 2, which reveals yet more topological detail.

It is safe to assume that the more reference points are used to cross-reference the delay map, the more topological details are revealed. there is, however, a physical limit to this process. From many simulations conducted using multiple reference points it was concluded that past 50% of all nodes used as reference points the topological view of the network degrades and starts to resemble the original geographical mesh. Ideally, if all nodes cross-reference each other, no topological information can be inferred from the results, as was already mentioned in the introduction.

It is important to note that Case 3 in Fig.5 is the closest view to the one presented earlier in Fig.4 given the number of branches from the virtual center of the community network. The only difference is that delays in Fig.5 are relative to reference points selected in the network to perform delay cross-reference. However, the target of this paper is not to pinpoint Euclidean coordinates of

individual nodes but to detect details of logical topology of the entire community network. Based on Fig.5 it is clear that the target has been reached.

4 Conclusions

This paper proposed to use a simple algorithm based on end-to-end delay measurements to infer logical topology in large-scale community networks. The key assumption of the proposed algorithm is that all networks have multiple-tiers and when a community network in question is global, end-to-end paths are destined to move up and down in the tier depth.

Another basic assumption is concerned with the precision of the proposed algorithm. The perfect precision is attainable only when there are as many reference points in the network as there are main branches in logical topology. However, even a few reference points are capable to creating visual differences when translated into cross-referenced delay maps.

Main targets of the proposed algorithm are community networks created on-demand with a relatively short lifespan. The opposite case would be a community network started from a single node incrementally adding more nodes as new members join the network. In this case the proposed algorithm should be refined since new members normally form a long-term relationship and there is time to perform a more time-consuming algorithm of topological inference. Vivaldi and Meridian models were created specifically for such incremental processes. With short-term global community networks this approach is not feasible, especially given that the emergence of NGN services will add a large community of mobile users that will be difficult to locate geographically.

The proposed algorithm assumes that higher delay distances in any global network occur from East to West or visa versa. This is true for contemporary networks which form a virtual belt around the globe. Based on this assumption, cross-referencing of delays is performed by using multiple horizontal points rather than trying to split the vertical scale. Although the concepts of horizontal and vertical bears little meaning when translated to network-wide connectivity, in practice this seems to be the local choice of words.

In several cases of delay-based topological inference it was established that the more mid-network reference points are used for cross-referencing end-to-end delays the more topological detail is detected in the resulting mapping. Although practical use cases of the proposed algorithm are not included in this paper, it is fair to assume that depending on each particular community network the number of cross-reference points should vary based on the required level of detail and resources allocated to topological inference.

In the continuation of this work, extensive tests are planned with the use of actual topologies obtained from publicly available repositories. The obtained results will be used to refine the algorithm as well as to create a universal model based on traditional graph theory.

References

1. GT-ITM: Georgia Tech Internetwork Topology Generator, <http://www.cc.gatech.edu/fac/Ellen.Zegura/graphs.html>
2. Tang, L., Zhang, H., Li, J., Li, Y.: End-to-End Delay Behavior in the Internet. In: 14th IEEE international Symposium on Modeling, Analysis, and Simulation (MASCOTS), pp. 373–382 (2006)
3. PlanetLab, <http://www.planet-lab.org>
4. Zeitoun, A., Chuah, C.N., Bhattacharrya, S., Diot, C.: An AS-Level Study of Internet Path Delay Characteristics. In: IEEE Globecom (2004)
5. Dabek, F., Cox, R., Kaashoek, F., Morris, R.: Vivaldi: a decentralized network coordinate system. SIGCOMM Computer Communications Review 34(4), 15–26 (2004)
6. Wong, B., Slivkins, A., Sirer, E.G.: Meridian: a lightweight network location service without virtual coordinates. In: Conference on Applications, Technologies, Architectures, and Protocols For Computer Communications, pp. 85–96 (2005)
7. p2psim, <http://www.pdos.lcs.mit.edu/p2psim/>
8. Zhang, B., Ng, T.E., Nandi, A., Riedi, R., Druschel, P., Wang, G.: Measurement based analysis, modeling, and synthesis of the internet delay space. In: ACM SIGCOMM Conference on internet Measurement, pp. 85–98 (2006)

Indexing through Querying in Unstructured Peer-to-Peer Overlay Networks

K. Haribabu¹, Chittaranjan Hota², and Antti Ylä-Jääski³

¹ Computer Sc. & Information Systems Group, Birla Institute of Technology and Science
Pilani, Rajasthan, India

² Computer Sc. & Information Syst. Group, Birla Institute of Technology and Science, Pilani
Hyderabad Campus, Hyderabad, Andhra Pradesh, India

³ Data Communications Software Laboratory, Department of Computer Sc. & Engineering
Helsinki University of Technology, Espoo, Finland
khari@bits-pilani.ac.in, hota@bits-hyderabad.ac.in,
antii.yla-jaaski@hut.fi

Abstract. The efficiency of a Peer-to-Peer file sharing overlay is measured in terms of the scalability and versatility of its object lookup strategy. In these networks peers carry out distributed query relaying to discover the service providers. Existing lookup mechanisms like flooding and random walks in unstructured P2P overlays create huge communication overhead and increased response time. In this work we propose efficient lookup in unstructured peer-to-peer overlay networks using indexing through querying, distributing indices through queries. Our simulation studies show that by our approach more than 97% of the queries are answered in one hop and the rest in few hops thus reducing the network load. Our approach is efficient in worst case scenarios where contents are distributed over thousands of peers and the overlay network condition is highly dynamic.

Keywords: Peer-to-Peer, Overlay Networks, Indexing, Lookup, Bloom Filters.

1 Introduction

The emergence of Peer-to-Peer (P2P) overlay file sharing networks has increased the interest amongst the Internet users to use the Internet beyond web browsing and exchanging e-mails. These networks have recently gained a lot of attention in academic and industrial circles. The P2P traffic occupies major chunk of the Internet traffic [1]. The kind of impact they generate on the whole Internet motivates the researchers to study and improve the scalability and performance of these networks.

P2P overlay networks are application-level logical networks built on top of the physical networks as shown in Fig 1. These networks maintain separate addressing and routing mechanisms to enable efficient search and data exchange between peers. They don't require any special administrative or financial arrangement. They are self-organizing and adaptive, distributed and decentralized. There are two kinds of requests that are sent over the overlay edges, i.e. content requests to find an object and

keep-alive requests to stay connected in the overlay. P2P overlay networks are categorized as unstructured and structured. An unstructured P2P system is composed of peers joining the network with some loose rules, without any prior knowledge of the topology. Gnutella [2], and KaZaA [3] are examples of unstructured P2P overlay networks. In structured P2P overlay networks, network topology is tightly controlled and content is placed not at random peers but at specified locations that will make subsequent queries more efficient. Most of the structured P2P overlays are Distributed Hash Table (DHT) based. Content Addressable Network (CAN) [4], Chord [5], and Pastry [6] are some examples of structured P2P overlay networks.

In unstructured P2P networks, the lookup for an object involves forwarding the queries to all neighbors [7]. This generates huge message traffic in the network. As the queries are forwarded, the traffic exponentially multiplies at each hop. This degrades the network performance thus limits the scalability in terms of number of queries. As the number of queries increase in the network, the whole network bandwidth is consumed by these query messages, thus burdening the peers with unnecessary duplicate messages, message processing and giving very huge response times to the querying peers.

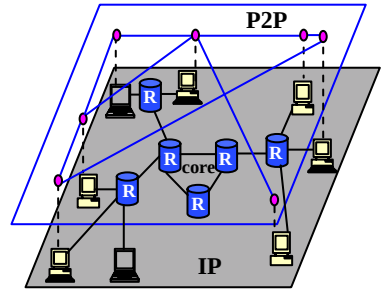


Fig. 1. An example Overlay Network

This has prompted us to think of ways to utilize the huge traffic generated and reduce further the query explosion. In this paper we present a query-based index propagation approach for disseminating the content indices through the query traffic. Each index record is the collection of bits generated by Bloom Filter [8]. Every query carries a summarized index of contents of the peers that are on its way. At every node, the indices are copied to local cache and the local indices are updated in the query message. The nodes can lookup the local index cache and directly contact the node for the object. If the index record is not found, it uses the usual lookup algorithm. Through the traffic the indexes are disseminated. As the time passes, the query traffic reduces and reaches a steady state.

The approach takes advantage of huge traffic for a good purpose. It offers many advantages. First it reduces the aggregate bandwidth consumption in the network. This improves the scalability. Second, it reduces the search latency by maintaining an index of the objects in the neighborhood. This improves the quality of the search giving satisfaction to the user. Third, the nodes which participate in processing the maximum traffic get the maximum benefit. Such nodes get one-hop access to the most of objects in the network. Fourth, the index is disseminated equally for both popular as well as unpopular objects. One of the demerits of unstructured networks is that it doesn't guarantee retrieval of an object although it is present in the network. This is due to few replicas of unpopular objects in the network. Due to this index distribution, many pointers to such objects are distributed thus reducing the risk of not finding objects. However, we do not address the free riding problem in P2P overlays in this work.

The simulation results show that Indexing Through Querying (ITQ) approach compared with Flooding and k-random walk techniques reduce the average bandwidth consumption and search latency by more than 95%. However, the ITQ approach is less efficient when there is very less query traffic in the system and there are very few replicas of the objects, which is of course an unlike scenario in today's peer-to-peer file sharing systems.

2 Related Work

Gnutella [2] used query flooding for locating the objects which obstructs the scalability of any network. In local indices technique [9], each node maintains an index of data of all nodes within r hops of itself. When a node receives the query, it processes it on behalf of all the nodes within r hops of radius. Creating and maintaining such index involves extra overhead on the system. In routing indices technique [10], index is created for different topics in different routes. The index is used for choosing a neighbor to forward the query. In this technique, the aggregate updates are exchanged among the nodes to keep the index up to date. By sending only the aggregated vectors, the overhead is reduced. In attenuated bloom filter technique [11], an index for every neighbor and up to d number of hops is maintained. Index for neighbor n , stores the bits of objects generated by Bloom Filter [8] available at hop d through neighbor n . The index is maintained by exchanging the changed bits. In [12], a global but partial index is built using a Distributed Hash Tables (DHT) built on top of the unstructured overlay. The index consists of peer's top interests which is reflected by the nature of queries and objects peer has. Changes in the interests of peers involve changing the neighbors. In [13], super peers develop multi-attribute based content index by observing the QueryHit messages that pass through it. They use this index to efficiently forward the queries to selected neighboring super peers. There is separate index table for every attribute. The entries are deleted based on age factor. In advertisement based technique [14], a local index is built by requesting neighbors up to certain hop level to send their ads or indexes. In eSearch [15], index for every term is created. One node is responsible for maintaining an index of one term. Nodes analyze their documents and find top terms and publish them to the respective nodes responsible for those terms.

Our approach differs from above that the index is propagated by the queries. Nodes have passive involvement of loading the indices to and from queries. So the index distribution is fully dependent on query distribution.

3 Indexing through Querying

In this section we discuss design of our approach.

3.1 Scope

In this paper we focus on unstructured peer-to-peer networks. The approach encourages the idea that instead of creating and maintaining indexes by individual nodes, index information is disseminated by the query traffic. Each node observes the

query traffic that passes through it and updates its local index cache. There is always a trade-off between the scope of the index and maintenance cost. In one extreme every node can keep index of all the objects available in the network. That will surely make all the queries *one-hop* (zero hops for lookup and one hop for getting the object). But the cost associated with maintaining such an index prohibits such an approach. On the other extreme searching loses its grace without indexing. Indexing is the most important tool for searching [16]. Our approach is based on day-to-day observation that information about festivals, events and government policies are propagated by sticking the posters on the public transportation buses, trains etc. The vehicles wherever they go, by their presence they communicate the information to the people. This doesn't cost extra. We augment that idea to suit the peer-to-peer networks. Each query can be seen as vehicle carrying the index information of all the nodes it has passed through. The nodes receiving the query copy all the index information and add their indices to the query. This way whoever the query reaches gets benefited. When the node needs to find the object, it looks up its index cache and if there is match it will immediately contact the peer having that object.

Our approach is fundamentally based on the observation that query traffic of public file sharing p2p networks occupies most of the internet traffic. In [1], it is observed that p2p network traffic occupies the 40% of total internet traffic. In [17], it is reported that there are 562 transactions per second happening in Kazaa [3] network. This clearly gives the extent of queries in p2p systems. Not only the successful queries but even the unsuccessful queries can be used as vehicles for carrying the index information. Not only the queries but even the query replies and download requests can be used for this purpose. That way we have huge percentage of traffic at our disposal for conveying. But in this paper we focus on query messages.

3.2 Data Structure

The query passes through many nodes and if every node keeps its index in the query, query will become bulky and will consume huge bandwidth. To address this issue we use Bloom Filters to only store the summarized index for specified number of files in one node. Bloom Filters [14] are widely used to represent elements of a set using a bit set hashed through a set of hash functions. The filter may report an element is present even though it is not present in the set. This is known as false positive. But the filter never reports absence of an element when it is present in the set. The false positive probability (*fpp*) of representing set of elements of size n depends on size m of Bloom Filter and size k of set of hash functions. From (2), it can be noted that *fpp* reduces exponentially as n is reduced, and k and m are increased. The m and *fpp* are computed as follows [26].

$$m = (k * n) / \ln 2 \quad (1)$$

$$fpp = (0.6185)^{m/n} \quad (2)$$

We have set of constraints when we apply Bloom Filters to our approach. First, query message size is limited. Maximum query size as per Gnutella protocol 0.6 [20], is 4 KB. The number of files shared by users varies from zero to thousands. So to have one bloom filter for all files in a node is not feasible. Second, Bloom Filters don't

have an option of deleting elements. It can only add new elements into the filter. So nodes exchange indices in units of Bloom Filter not a part of it. It requires identifying each Bloom Filter uniquely across the network so that it is possible to replace old Bloom Filter with new one.

As observed in [18], in a p2p file sharing network, 75% of nodes share 100 or less number of files. So it will be appropriate to group 100 files of a node and represent their file names in one Bloom Filter. Nodes having more than 100 files will have more than one Bloom Filter to convey their indices. By fixing size of n at 100, we have various alternatives in choosing size of m and k as listed in Table 1. As k increases fpp falls exponentially. The choice of k effects the bandwidth consumption, and the time required to disseminate the indices.

Table 1. k and m and fpp

k	fpp	m (bytes)
3	0.125	54
4	0.0625	72
5	0.03125	91
6	0.01561	109
7	0.00781	126
8	0.00389	144
9	0.001945	162

For supporting efficient dissemination and updating of indices across the network without extra cost we propose the following structure called Query Index Record (*QIR*). *QIR* record is denoted as $(N, B, \{R, S, D_c\})$. N refers to the node identity such as IP address and B refers to the bandwidth (mbps) of the connection available at node N , and rest is the set of Bloom Filter records (*BFR*) of N . Each *BFR* has R , the record number unique for N , S refers to m bit array, and D_c refers to the date on which this bloom filter is created by N . It is measured as seconds that have elapsed since the starting of the network. This date is used to compare the freshness of the *BFR*. A *BFR* is uniquely identified by the combination of N and R . A query carries several *QIR* records as many as it can fit in maximum message size. As observed in [18], a node is sharing maximum of 8000 files. Each *BFR* can store summary of 100 files as discussed above. The value of R doesn't exceed 100 in worst case thus takes 1 byte space. For $k=6$, the size of *BFR* record in bytes is $(1, 109, 4) = 114$ bytes. The size of *QIR* record in bytes is $(4, 1, \text{size of set of } BFR)$. Also each node has a local cache of Node index records (*NIR*). The structure of *NIR* is $(N, B, \{R, S, D_c, D_{lu}, D_{li}\})$. For each *BFR*, node stores additional information like the date on which it was last successfully used (D_{lu}), and the date on which it was loaded on to a query (D_{li}). D_{lu} is used for removing the *NIR* after certain time and D_{li} is used in deciding which indices to be loaded onto the query.

Let us apply the above structure to Gnutella 0.6 [20] with $k=6$. A Gnutella Query message takes maximum of 54 bytes assuming that keywords usually take less than 30 characters thus maximum of 30 bytes. Maximum message size is 4 KB. Then it is possible to fit in $(4096 - 54) / 119 = 35$ index records in each Query message. The 35 index records convey the locations of 3500 objects in the network. What should be done when there are more than 35 index records to load on to a query is discussed in next section.

3.3 Algorithm Overview

When a node joins the network, during the initial handshake with neighbors, it collects the set of *NIR* from each neighbor and updates its local index cache. When a

node wants to lookup for set of keywords W , it first looks up its own files and index as shown in the search algorithm in Fig 2. Out of set of hosts H , *choseHostWithMaxBw* procedure chooses the host h with maximum bandwidth. If there are no matches, it will resort to usual lookup method i.e, flooding or random walk. The query Q is populated with the cached indices. In case the node has more indices than it could fit in the query, it will choose the indices based on $D_{||}$ as shown in Fig 3. The indices are chosen in a round-robin fashion ensuring that all the indices are propagated. The *sort* method sorts the index records on $D_{||}$ in ascending order. Similarly when a node receives the query, it will scan the query Q for the *QIR* records and either replaces or adds them to local cache. The criterion for either replacing or adding is shown in Fig 4. After updating the local index cache, it will search files and the index cache similar to search algorithm in Fig 2. If it finds a match(s), it will make reply message and includes the addresses and the bandwidth information of all the nodes it found to have the desired object. But it doesn't confirm whether the object really exists in the network. So the result may include the false positives also. If it doesn't find a match, it will forward the query to its neighbors.

```

Search(Keywords:  $W$ )
{
  Result:  $R$ 
  Host:  $H$ 
  for each file  $f \in F$ 
    if  $match(f, W) = \text{true}$  then
       $R \leftarrow R \cup f$ 
  If  $R = \Phi$  then
    for each index record  $i \in I$ 
      if  $BloomFilterMatch(i, W)$  then
         $H \leftarrow H \cup (N(i), B(i))$ 
  else return  $R$ 
  If  $H = \Phi$  then
     $Q \leftarrow makeQuery(W)$ 
     $Q \leftarrow loadIndex(Q, I)$ 
    for each neighbor  $n \in P$ 
      forwardQuery( $n, Q$ )
  else
     $h \leftarrow choseHostWithMaxBw(H)$ 
    requestDownload( $h, W$ )
}

```

Fig. 2. Search algorithm for searching W at a querying node with index cache I , set of Files F and set of neighbors P

```

LoadIndex(Query:  $Q$ , Index:  $I$ )
{
  number:  $QSize = 0$ 
   $I \leftarrow sort(I, "D_{||}")$  //sorts  $I$  on  $D_{||}$ 
   $Q \leftarrow clearIndex(Q)$ 
  for each index record  $i \in I$ 
     $Q \leftarrow load(Q, N(i), B(i), R(i), S(i), D_c(i))$ 
     $D_{||}(i) \leftarrow now()$ 
     $QSize \leftarrow QSize + Z$ 
  if  $QSize + Z > M$  then
    return }

```

Fig. 3. Algorithm for adding index records in Query Q at a node with index cache I and network wide maximum query size M and query index record size Z

```

UpdateIndexInNode(Query:  $Q$ , Index:  $I$ )
{
  boolean  $found$ 
  for each index record  $r \in R$ 
     $found \leftarrow false$ 
  for each index record  $i \in I$ 
    if  $N(i) = N(r)$  and  $R(i) = R(r)$  then
       $found \leftarrow true$ 
      if  $D_c(i) < D_c(r)$  then
         $i \leftarrow r$ 
  if  $found = false$  then
     $I \leftarrow I \cup r$  }

```

Fig. 4. Algorithm for adding index record set R of Query Q to a node's index cache I

4 Simulation Results

We used a 1000 node random graph generated by GT-ITM [21] with average degree of 10. We used a pure P2P model where there is no distinction among the peers. All peers have equal functionalities. There are 500 distinct objects in the network with the 1st object being the most popular and 500th object being the least popular. The objects and queries are distributed according to the observations made in [22]. The replica distribution of these objects is done according to the Zifian distribution with parameter $\alpha = 0.82$. The replicas are placed at randomly selected nodes. Queries for these objects also follow the Zifian distribution ($\alpha = 0.82$) i.e. the popular objects receive more queries than that of less popular objects. The queries are generated according to a Poisson process with average generating rate at 8 queries per second. For each query, a node is selected randomly to fire that query. The simulation was run for 7500 queries. To model the churn scenario, we remove 10% active nodes and insert equal number of new nodes into the system. The new nodes start functioning without any prior knowledge. The objects are replicated onto these new nodes according to Zifian distribution. To model the object deletions and insertions, randomly selected objects are removed from randomly selected nodes and new replicas and new objects are added at randomly selected nodes. The topology change happens every 3 minutes and object relocation happens every 3 minutes during each run. For simulating the search method flooding, the TTL is set to 7 and for k -random walk 5 walkers are used and TTL for each is set to 1024. The parameters for Bloom Filter are number of files $n= 10$, $k=6$ and maximum query size is 4096 bytes.

We simulate Flooding (FL), k -Random walk (RW), Index Dissemination with Flooding (ITQFL), Index dissemination with k -Random walk (ITQRW). The comparison is done in terms of bandwidth consumption, query response time, and false positives. In Fig 5, we show the average messages produced per query. There is 99% reduction in average messages in ITQ approaches. The difference between flooding and random walk is due to forwarding only to few randomly selected neighbors. But the huge difference between ITQ and others is due to the spreading of the object indices through the queries. By spreading indices, the location information of the objects is available within fewer hops of the requesting nodes. As the query is answered within fewer hops, correspondingly the message replication also reduces exponentially. In Fig 6, we compare index propagation through random walk and flooding. In both the approaches, the curve rises steeply for the first few seconds and rises slow after that. This is because; initially both approaches behave just like their counterparts. But the difference is that the initial query traffic carries the indices and spreads to all the nodes on its way. So there is decline in steepness of the curves. There are rises in the steepness of the curve at 184, 385, and 522 seconds indicating the churn scenarios. But it can be clearly seen that the rise in the steepness reduces over next few seconds and becomes normal. Thus our approach adapts to the churn scenarios and stabilizes the network.

As k increases, the false positive rate reduces and the length of Bloom Filter or BFR record increases thus loading fewer records onto the query. This can be observed in Fig 7 and 8. First in Fig 7 there is less query traffic in the starting of the network compared to Fig 8. This is attributed to more query hits that occur in the first few hops and they don't further forward the query. The more query hits are due to high

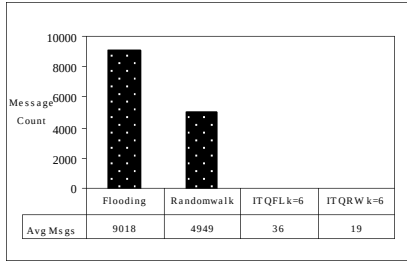


Fig. 5. Average messages per query

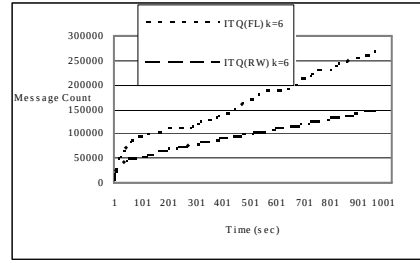


Fig. 6. CDF of bandwidth consumption

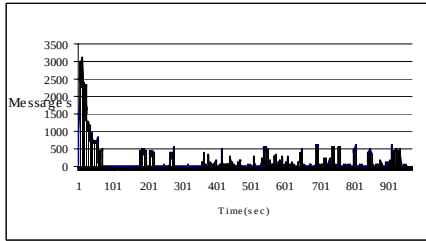


Fig. 7. Traffic distribution for ITQFL k=3

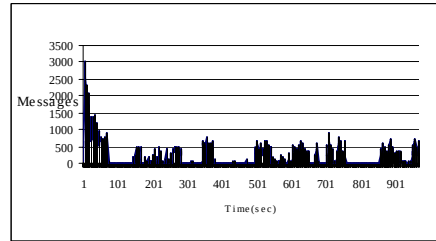


Fig. 8. Traffic distribution for ITQFL k=7

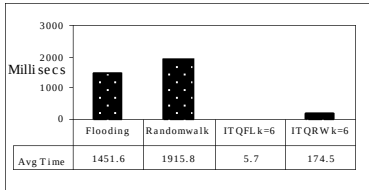


Fig. 9. Comparison of avg response time

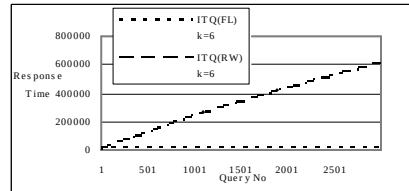


Fig. 10. CDF of query response time

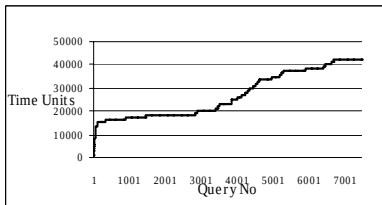


Fig. 11. CDF of response time (ITQFL k=6)

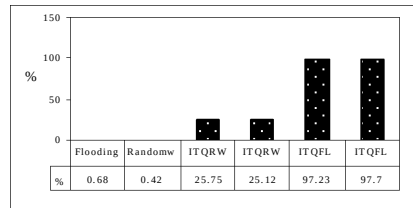


Fig. 12. Percent queries answered in one-hop

false positive rate. In Table 2, we observe that there are on an average 33.6 false results for every query. As k , increases the false positives decrease considerably. Second as k increases, the number of indices that can be loaded onto the query reduce

linearly. Thus it takes time to stabilize the network. The comparison of average query response time is depicted in Fig 9. We see that random walk has maximum search latency. This is attributed to depth first search. There is drastic reduction in search latencies of ITQ approaches when compared to others. The comparison between search latencies of ITQRW and ITQFL is presented in Fig 10. The difference in steepness is due to, first, in random walk, the indices are spread not breadth wise but depth wise. So only few nodes come in contact with the queries. Second the nodes are selected randomly to forward the query. This means a node may be selected several times leaving other nodes not getting selected at all. This way the network has rare chance of getting stabilized. The churn further complicates this. Although in Fig 10, the curve for ITQFL seems to be constant, but if we closely look at it, it looks as in Fig 11. For most of the queries, the curve is flat, indicating that queries are answered in *one-hop*. The rises in steepness of the curve are due to the churn. But soon the curve becomes flat indicating that the network has recovered from the churn. The percentage of queries answered with the locally available index (*one-hop queries*) is shown in Fig 12.

Table 2. False positives per query in ITQFL

k	fp
3	33.63
4	28.19
5	25.7
6	24
7	21.65

5 Conclusion

In this paper we propose a novel approach, indexing through querying that utilizes the huge amount of traffic to propagate the indices of the objects. Our approach utilizes the benefits of query forwarding to improve upon the scalability of P2P lookup. It also improves upon the availability in the face of individual failures through index replication. Experimental evaluations have shown that our approach works very efficiently in conjunction with flooding in improving the search efficiency. The search cost is reduced over by 95% and over 97% queries were answered *one-hop*. We plan to address the free riding problem in the future work.

References

1. Saroiu, S., Gummadi, K., Dunn, R., Gribble, S., Levy, H.: An Analysis of Internet Content Delivery Systems. In: Proceedings of the Fifth Symposium on Operating Systems Design and Implementation, USENIX Association, Berkeley, CA, USA, pp. 315–328 (2002)
2. Gnutella Protocol Specification Version 0.4, http://www9.limewire.com/developer/gnutella_protocol_0.4.pdf
3. Kazaa, <http://www.kazaa.com>
4. Ratnasamy, S., Francis, P., Handley, M., Karp, R., Shenker, S.: A Scalable Content Addressable Network. In: Proceedings of the 2001 ACM Annual Conference of the Special Interest Group on Data Communication (SIGCOMM), pp. 161–172. ACM Press, New York (2001)
5. Stoica, I., Morris, R., Liben-Nowell, D., Karger, D., Kaashoek, M.F., Dabek, F., Balakrishnan, H.: Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications. IEEE/ACM Transactions on Networking 11, 17–32 (2003)

6. Rowstron, A., Druschel, P.: Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems. In: Proceedings of IFIP/ACM International Conference on Distributed Systems Platforms (Middleware), London, pp. 329–350. Springer, Heidelberg (2001)
7. Yang, B., Patrick, V., Garcia-Molina, H.: Evaluating GUESS and Non-Forwarding Peer-to-Peer Search. In: Proceedings of IEEE 24th International Conference on Distributed Computing Systems (ICDCS), pp. 209–218. IEEE Computer Society, Washington (2004)
8. Bloom, B.H.: Space/time trade offs in hash coding with allowable errors. *Communications of the ACM* 13(7), 422–426 (1970)
9. Yang, B., Garcia-Molina, H.: Improving search in peer-to-peer networks. In: Proceedings of the 22nd International Conference on Distributed Computing Systems (ICDCS), p. 5. IEEE Computer Society, Washington (2002)
10. Crespo, A., Garcia-Molina, H.: Routing indices for peer-to-peer systems. In: Proceedings of the 22nd International Conference on Distributed Computing Systems (ICDCS), pp. 23–32. IEEE Computer Society, Washington (2002)
11. Sean, C.R., Kubiawicz, J.: Probabilistic location and routing. In: Proceedings of 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2002), pp. 1248–1257. IEEE Computer Society, Washington (2002)
12. Zhang, R., Hu, Y.C.: Assisted Peer-to-Peer Search with Partial Indexing. *IEEE Transactions on Parallel and Distributed Systems* 18, 1146–1158 (2007)
13. Praveen, K., Srividya, G., Sridhar, V.: Multi-attribute Based Similar Content Indexing in Hybrid Peer-to-Peer Networks. In: Proceedings of the International conference on Networking and Services (ICNS), p. 25. IEEE Computer Society, Los Alamitos (2006)
14. Peng, G., Jun, W., Hailong, C.: ASAP: An Advertisement-based Search Algorithm for Unstructured Peer-to-peer Systems. In: Proceedings of the 2007 International Conference on Parallel Processing (ICPP), p. 8. IEEE Computer Society, Washington (2007)
15. Tang, C., Dwarkadas, S.: Hybrid global-local indexing for efficient peer-to-peer information retrieval. In: Proceedings of the 1st Symposium on Networked Systems Design and Implementation, USENIX Association Berkeley, CA, USA, p. 16 (2004)
16. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison Wesley Longman, Boston (1999)
17. Gummadi, K.P., Dunn, R.J., Saroiu, S., Gribble, S.D., Levy, H.M., Zahorjan, J.: Measurement, Modeling, and Analysis of a Peer-to-Peer File-Sharing Workload. *ACM SIGOPS Operating Systems Review* 37, 314–329 (2003)
18. Saroiu, S., Gummadi, P.K., Gribble, S.D.: A measurement study of peer-to-peer file sharing systems. In: Proceedings of Multimedia Computing and Networking (MMCN) (2002)
19. Fan, L., Cao, P., Almeida, J., Broder, A.: Summary Cache: A Scalable Wide-Area Web Cache Sharing Protocol. *IEEE/ACM Transactions on Networking* 8, 281–293 (2000)
20. Gnutella protocol 0.6, http://rfc-gnutella.sourceforge.net/src/rfc-0_6-draft.html
21. Zegura, E.W., Calvert, K.L., Bhattacharjee, S.: How to Model an Internetwork. In: Proceedings of Fifteenth Annual Joint Conference of the IEEE Computer Societies, pp. 594–602. IEEE Computer Society, New York (1996)
22. Chu, J., Labonte, K., Levine, B.: Availability and Locality Measurements of Peer-to-Peer File Systems. In: Proceedings of SPIE (2002)

TrustRRep: An Improved Reputation Management Scheme for Reliable Resource Sharing in Peer-to-Peer Networks

Junghwa Shin, Taehoon Kim, and Sungwoo Tak*

School of Computer Science and Engineering, Pusan National University,
San-30, Jangjeon-dong, Geumjeong-gu,
Busan, 609-735, Republic of Korea
shinjh@pknu.ac.kr, ninkth@hotmail.com, swtak@pusan.ac.kr

Abstract. Inauthentic resources can be easily spread by peers due to the openness and anonymity of P2P networks. A possible way to restrict the distribution of inauthentic resources and prevent malicious peers from joining P2P networks is to exploit a reputation which reflects peers' past behaviors. There is a possibility that a peer intentionally plays along with other peers in order to increase/decrease its reputation through dishonest feedback exchanges. Therefore, we propose a new reputation management scheme, called TrustRRep (**Trustworthy Resource sharing using Reputation**), which improves the trustworthiness and efficiency of P2P networks. We implemented the TrustRRep scheme on the NS-2 simulator for evaluating its performance compared to the recent reputation management work available in literature. A case study on simulations shows that the TrustRRep scheme yields efficient performance in terms of the minimal download ratio of inauthentic resources, the computation error of trust value and the success ratio of download request.

Keywords: P2P, Reputation, Dishonest feedback.

1 Introduction

In P2P networks, peers can spread the inauthentic resources or low quality resources easily [1]. To cope with the dissemination of inauthentic resources in P2P networks, a reputation that reflects past behaviors of peers is required. A reputation is computed by accumulating the feedback received from all other peers who have interacted with a peer and is helpful to predict peers' future behaviors. Besides, a reputation can provides shared resources and peers with trustworthiness, but this value is computed without consideration on the correctness of feedback. Under the deployment of reputation, there is a possibility that a peer intentionally plays along with other peers in order to increase/decrease its reputation through dishonest feedback exchanges [2]. Therefore, we propose a new reputation management scheme, called TrustRRep (**Trustworthy Resource sharing using Reputation**), which improves the trustworthiness

* Corresponding author.

and efficiency of P2P networks by identifying peers who give dishonest feedback. The TrustRRep scheme is also capable of providing peers with the trustworthiness on shared resources by discriminating inauthentic resources distributed by malicious peers. We implemented the TrustRRep scheme on the NS-2 Simulator [3] for evaluating its performance compared to the recent reputation management work available in literature. A case study on simulations shows that the TrustRRep scheme yields efficient performance in terms of the minimal download ratio of inauthentic resources, the computation error of trust value and the success ratio of download request. It also shows that the TrustRRep scheme imposes the restriction on participation of malicious peer by diminishing its trust value. The rest of this paper is organized as follows. Section 2 investigates related work in the research area of reputation management scheme in P2P networks. Section 3 presents the TrustRRep scheme that can identify peers who give dishonest feedback as well as provide shared resources and peers with trustworthiness. Section 4 analyzes the performance of TrustRRep scheme. Section 5 concludes this paper.

2 Related Work

P2P networks utilize mainly a reputation to provide shared resources and peers with trustworthiness, many schemes available in literature have been proposed to manage of reputation. The EigenTrust [4] scheme is the reputation scheme that decreases the number of malicious peers who spread the inauthentic resources. This scheme restrains the participation of malicious peers by referring a reputation of peer, but does not identify peers who give dishonest feedback. The PeerTrust [5] scheme computes the credibility value of peer by computing the similarity of reputation peers exchange. Also, this scheme assigns incentive to peers who give a feedback to a target peer, but has no consideration on the correctness of feedback. The FileTrust [6] scheme classifies reputation objects into a shared resource and a peer respectively. This scheme selects the resource with the highest reputation first and selects the peer with the lowest contribution value among peers who provide the resource. Therefore, the load can be shared by peers except high reputable ones. The Iguchi [7] scheme manages a reputation using a central server in unstructured P2P network and classifies reputation objects into a resource and a peer respectively. This scheme selects the resource with the highest reputation value and selects the peer with the highest contribution value among peers who provide the resource.

3 An Improved Reputation Management Scheme for Reliable Resource Sharing in Peer-to-Peer Networks

In this section, we describe the TrustRRep scheme that provides shared resources and peers with trustworthiness and improves the trustworthiness and efficiency of P2P networks by identifying peers who give dishonest feedback.

3.1 Characteristics of TrustRRep Scheme

The existing reputation management schemes are subject to provide shared resources and peers with trustworthiness, having no consideration for identification of peers who give dishonest feedback. Peers can download incorrect resources or inauthentic resources by peers who give dishonest feedback. Because peers generally select a resource referring the feedback provided by other peers, the correctness of feedback provided by peers is very important. However, sometimes peers intentionally play along with other peers in order to increase/decrease its reputation through dishonest feedback exchanges. Basically, such a peer who gives dishonest feedback is referred to as a liar. A dishonest feedback of peer has a chance of posing serious problems such as reputation increment of a malicious peer and reputation decrement of a good peer, thus, the identification on a liar needs to be carefully considered [8]. Our scheme improves the trustworthiness and efficiency of P2P networks by identifying peers who give dishonest feedback and by providing shared resources and peers with trustworthiness. Our scheme has the following characteristics. A resource reputation represents how trustworthy it is. A peer reputation represents how many trustworthy resources it has contributed. And then this value is computed with the sum of all shared resources' reputation. Additionally, we compute a credibility ratio and a trust value of peer. A credibility ratio is used to identify whether a peer is liar or not. We exploit a trust value of peer as a measurement for selecting a peer. There are a number of pre-trusted peers to manage information such as shared resource lists of peer, a reputation of resource, a reputation of peer, a trust value of peer and a credibility ratio of peer. Peers ask pre-trusted peers for relevant information to select a resource and to respond the download request of peers. Peers also submit a feedback on downloaded resource to pre-trusted peers after the transaction. We choose initially a pre-trusted peer based on the number of shared resources per peer. After that, we choose a peer with the highest trust value as pre-trust peer. A pre-trust peer may continuously change as transactions are increased. The procedure of TrustRRep scheme consists of 4 steps: (1) *Join*, (2) *Query* and *QueryHit*, (3) *Selection* and *Download*, (4) *Evaluation* and *Update*.

3.2 Reputation Management of TrustRRep Scheme

Here are the parameters for describing the overall procedure of the TrustRRep scheme.

We now describe the detailed procedure of TrustRRep scheme exploited in P2P networks.

Procedure I (Join) represents a procedure that peers participate in P2P networks and pre-trust peers initialize. Each peer sends shared resource lists to a pre-trusted peer. The pre-trusted peer initially assigns a value of 1 to a reputation of resources (*RR*), a reputation of peers (*PR*), a trust value of peers (*T*) and a credibility ratio of peers (δ).

Procedure II. Query and QueryHit

```

(Step 1)  $P_A$  : Query
  for all  $Neighbors(P_A)$  do
    send Query <KEY,  $TTL_{init}$ ,  $P_A$ > to  $Neighbors(P_A)$ 
  end-loop
(Step 2)  $P_j \in Neighbors(P_A)$  : QueryHit
  if find(KEY) != NULL then

```

for all $j=1,\dots,n$ **do**
 send $QueryHit < P_j, r_{P_j}^1, \dots, r_{P_j}^m >$ to P_A (2-1)

end-loop

else

$TTL \leftarrow TTL-1$ (2-2)

if $TTL > 0$ **then**

for all $Neighbors(P_j)$ **do** (2-3)

 send $Query < KEY, TTL, P_A >$ to $Neighbors(P_j)$

end-loop

end-if

end-if

Table 1. Parameters

Symbol	Description
P_i	Stands for the identity of peer, where $i=\{1,2,\dots,n\}$
$Neighbors(P_i)$	Represents a set of neighbor peers of peer P_i , where $i=\{1,2,\dots,n\}$
P_A	Resource requesting peer
P_B	Resource providing peer
$r_{P_i}^k$	Stands for the identify of resource that peer P_i shares, where $i=\{1,2,\dots,n\}$, $k=\{1,2,\dots,m\}$
$RR(r_{P_i}^k)$	Represents a reputation on resource of peer P_i , where $i=\{1,2,\dots,n\}$, $k=\{1,2,\dots,m\}$
$PR(P_i)$	Represents a reputation of peer P_i , where $i=\{1,2,\dots,n\}$
$\delta(P_i)$	Represents a credibility ratio of peer P_i , where $i=\{1,2,\dots,n\}$
$T(P_i)$	Represents a trust value of peer P_i , where $i=\{1,2,\dots,n\}$
KEY	Represents a search keyword
$FV_{P_i,P_j}(r_{P_j}^k)$	Stands for a feedback provided by P_i on resource of peer P_j , where $ij=\{1,2,\dots,n\}$, $k=\{1,2,\dots,m\}$
$\alpha(r_{P_i}^k)$	Represents the ratio of good feedback on resource of peer P_i , where $i=\{1,2,\dots,n\}$, $k=\{1,2,\dots,m\}$
$\beta(r_{P_i}^k)$	Represents the ratio of bad feedback on resource of peer P_i , where $i=\{1,2,\dots,n\}$, $k=\{1,2,\dots,m\}$
$CFN(P_i)$	Represents the number of correct feedback that peer P_i submits, where $i=\{1,2,\dots,n\}$
$TFN(P_i)$	Represents the number of total feedback that peer P_i submits, where $i=\{1,2,\dots,n\}$
$TDN(r_{P_i}^k)$	Represents the number of total download on resource of peer P_i , where $i=\{1,2,\dots,n\}$, $k=\{1,2,\dots,m\}$
$TN(P_i)$	Represents the number of total transactions of peer P_i , where $i=\{1,2,\dots,n\}$

Procedure II represents a procedure that peers send a *Query* message and receive a *QueryHit* message from peers having a resource. To obtain a resource, a resource requesting peer P_A sends a *Query* message to his neighbor peers (Step 1). A *KEY* represents a search keyword of resource. A TTL_{init} represents the number of neighbor peers who can receive the *Query* message. At each peer, the *TTL* value is decremented by 1. As soon as a peer sees a message with a *TTL* of 0, the message is dropped. The (Step 2) describes the behavior of neighbor peers who received the *Query* message. Among neighbor peers, if the peer P_j ($j=1,2,\dots,n$) has the resource that matches the *Query* message, he sends the his identity (P_j) and resource's identity to the peer P_A (Step 2-1). If the peer P_j has not the resource that matches the *Query* message, he decreases the *TTL* value by 1 and sends the *Query* message to his neighbor peers (Step 2-2 and Step 2-3). And then the peer P_A receives the *QueryHit* message from peers having the resource.

Procedure III. Selection and Download**(Step 1)** P_A : Request**for all** $i=1, \dots, n$ **do** send request $\langle RR(r_{P_i}^1), \dots, RR(r_{P_i}^m) \rangle$ to TP_i **end-loop** all $TP_i \in TP$: send response $\langle RR(r_{P_i}^1), \dots, RR(r_{P_i}^m) \rangle$ to P_A **(Step 2)** P_A : Select select $r_{P_B}^k$ with the highest RR among $\{RR(r_{P_i}^1), \dots, RR(r_{P_i}^m)\}$ send download request $\langle P_A, r_{P_B}^k \rangle$ to P_B **(Step 3)** P_B : Request**for all** $i=1, \dots, n$ **do** send request $\langle T(P_A) \rangle$ to TP_i **end-loop** all $TP_i \in TP$: send response $T(P_A)$ to P_B **(Step 4)** P_B : Receive receive response $T(P_A)$ **if** $T(P_A) \geq 0$ **then** upload $r_{P_B}^k$ to P_A **else** ignore P_A 's download request **end-if**

Procedure III represents a procedure that peers select a resource and request the download of resource. The peer P_A refers the *QueryHit* message received in the Procedure II, then it requests the reputation of resource to the pre-trusted peer TP_i . The pre-trusted peer TP_i sends corresponding information to the peer P_A (Step 1). The peer P_A selects the resource ($r_{P_B}^k$) with the highest resource reputation. Then it requests the download to the peer P_B having the resource (Step 2). Before the peer P_B responses the request of peer P_A , he requests the trust value ($T(P_A)$) of peer P_A to the pre-trusted peer TP_i . The pre-trusted peer TP_i sends the trust value of peer P_A to the peer P_B (Step 3). If the trust value of peer P_A is greater than 0, the peer P_B sends the resource to the peer P_A . Otherwise, the peer P_B ignores the request of peer P_A (Step 4).

Procedure IV. Evaluation and Update**(Step 1)** P_A : Send**for all** $i=1, \dots, n$ **do** send feedback $FV_{P_A, P_B}(r_{P_B}^k)$ to TP_i (feedback value : 1 or -1)**end-loop****(Step 2)** all $TP_i \in TP$: receive $FV_{P_A, P_B}(r_{P_B}^k)$ update δ, RR, PR, T

$$\delta(P_A) = \frac{CFN(P_A)}{TFN(P_A)}, 0 \leq \delta \leq 1 \quad (1)$$

$$RR(r_{P_B}^k) = \frac{1}{TDN(r_{P_B}^k)} \sum_{k=1}^{TDN(r_{P_B}^k)} FV_{P_A, P_B}(r_{P_B}^k) \quad (2)$$

$$PR(P_A) = \frac{1}{TN(P_A)} \sum_{k=1}^{TN(P_A)} RR(r_{P_A}^k) \quad (3)$$

$$T(P_A) = PR(P_A) \times \delta(P_A) \quad (4)$$

Procedure IV represents a procedure that the peer P_A evaluates the downloaded resource and the pre-trusted peer updates. The peer P_A submits the feedback on the download resource to the pre-trusted peer TP_i (Step 1). If the downloaded resource is identified the same resource as one the peer P_A asks for in Step 2 of Procedure III and is successfully executed, the peer P_A submits a value of 1 to the pre-trusted peer TP_i and otherwise a value of -1. If the peer P_A does not submit the feedback to the pre-trust peer TP_i , the pre-trusted peer TP_i assigns a value of 0 the peer P_B . The pre-trusted peer TP_i confirms the correctness on feedback of peer P_A and computes a credibility ratio ($\delta(P_A)$) of P_A . And then the pre-trusted peer TP_i identifies whether the peer P_A is a liar. In our scheme, the detailed procedures that identify whether a peer is a liar are as follows.

In the (Step 1), when the peer P_A submits the feedback of download resource to the pre-trusted peer TP_i , the pre-trusted peer TP_i computes the ratio of good feedback ($\alpha(r_{P_B}^k)$) and bad feedback ($\beta(r_{P_B}^k)$) among total feedback that the resource receives. To confirm the correctness on feedback of peer P_A , we used the probability variable X having the value from 0 to 1 as acceptance criteria like equation (5). The probability variable X follows the uniform distribution.

$$\alpha(r_{P_B}^k) + \beta(r_{P_B}^k) = 1, X \sim [0,1] \quad (5)$$

If the peer P_A submits the good feedback to the resource, the pre-trusted peer TP_i refers the probability variable X of equation (5) and confirms the correctness of feedback using equation (6). If the peer P_A submits the bad feedback to the resource, the pre-trusted peer TP_i refers the probability variable X of equation (5) and confirms the correctness of feedback using equation (7). In equation (6) and equation (7), the $FV(1)$ and $FV(-1)$ represents the feedback received from the peer P_A .

$$FV(1) = \begin{cases} TRUE, & \text{if } 0 \leq X \leq \alpha(r_{P_B}^k) \\ FALSE, & \text{if } \alpha(r_{P_B}^k) < X \leq \alpha(r_{P_B}^k) + \beta(r_{P_B}^k) \end{cases} \quad (6)$$

$$FV(-1) = \begin{cases} FALSE, & \text{if } 0 \leq X \leq \alpha(r_{P_B}^k) \\ TRUE, & \text{if } \alpha(r_{P_B}^k) < X \leq \alpha(r_{P_B}^k) + \beta(r_{P_B}^k) \end{cases} \quad (7)$$

The pre-trusted peer TP_i computes the number of total feedbacks ($TFN(P_A)$) submitted by the peer P_A and the number of correct feedbacks ($CFN(P_A)$) on the peer P_A according to the result of equation (6) and equation (7). Then the pre-trusted peer TP_i computes a credibility ratio of peer P_A using equation (1) and identifies whether the peer P_A is liar or not. In experimental of this paper, if the credibility ratio of peer P_A is less than 0.5, the pre-trusted peer TP_i decides on the peer P_A to be a liar. After the pre-trusted peer TP_i confirms whether the peer P_A is liar or not, it updates following values (Step 2).

A resource reputation is evaluated as described in equation (2). This value computes by the number of total downloads ($TDN(r_{P_B}^k)$) and the sum of feedback on the resource. Because the TrustRRep scheme computes the resource reputation except the feedback received from the liars, we can improve the trustworthiness of resource reputation. A reputation of peer is evaluated using equation (3). This value computes using the number of total transactions ($TN(P_A)$) and the sum of reputation on shared resource of the peer P_A . We finally evaluate a trust value of peer as described in equation (4) referring the credibility ratio of peer computed by equation (1) and the

reputation of peer computed by equation (3). If the peer P_A decided as a liar by equation (1), his credibility ratio is low though he provides authentic resources. Therefore, the trust value of peer P_A may decrease gradually. Besides, as the reputation of peer computed by equation (3) affected the trust value of peer, if the reputation of peer decreases the trust value of peer decreases too.

4 Experiments

In this paper, we compared the TrustRRep scheme with the work presented in EigenTrust, PeerTrust, FileTrust and Iguchi schemes under the following experimental environments to verify the efficiency of TrustRRep scheme. We have conducted experiments with the well-known NS-2 simulator [3] that empirically proves to simulate the right behavior patterns of peers in overlay networks running on the wired and wireless networks. Here are the simulation parameters.

Table 2. Simulation Parameters

	Description	Default
Network	# of peers in P2P networks	100
	# of resources initially distributed	400
	# of resources per a peer	$2 \leq n \leq 8$
	# of TTL	5
	# of max in-degree of a peer	4
	# of max out-degree of a peer	4
Peer's behavior	% of peers who distribute inauthentic resources	20%
	% of liar	20%
	# of new resources which malicious peers periodically publish	2
	% of response when a peer received the query	100%
Simulation	# of total transactions	5,000
	# of experiments over which results are averaged	10

In the experiments, we assign a value of 1 to initial values of the reputation on shared resources and peers, trust values and credibility ratio of peers. We use a value of 1 or -1 with feedback value of shared resource. Table 3 shows the four experimental cases. We simultaneously consider the behavior of malicious peer, the existence of liar and peers' rejoin.

In the first experiment, we confirmed the download ratio of inauthentic resources as transactions are increased. Fig. 1 shows the result of Case I.

Table 3. Experimental Cases

Case	Malicious peer's behavior	Liar	Rejoin
I	Upload inauthentic resources	No Liar	No admit
II		No Liar	Admit
III		Malicious peer is a liar	No admit
IV		Some peers give a false feedback	Admit

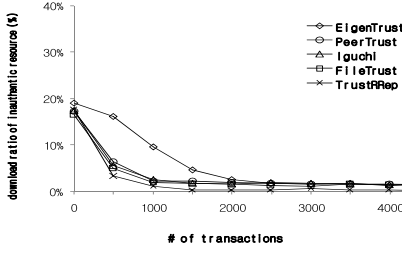


Fig. 1. Case I

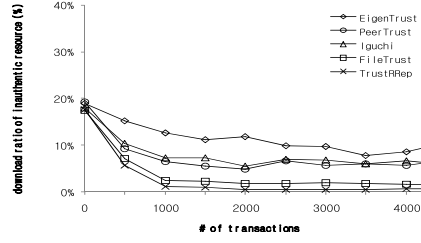


Fig. 2. Case II

As the EigenTrust scheme uses the probability selection method, it may choose a peer who has a relatively low trust value. This scheme shows a low performance with increment of download on inauthentic resources early. On the other hand, because other schemes including the TrustRRep scheme choose a resource or peer with the highest reputation or trust value, they have a similar performance. Fig. 2 shows the experimental result of Case II. The EigenTrust, PeerTrust and Iguchi schemes allocate default value that defined in the system to peers' reputation when peers rejoin. The TrustRRep scheme classifies reputation subjects into a resource and a peer respectively. Then the TrustRRep scheme computes the reputation of peer referring the resource reputation of equation (3) when peers rejoin. Therefore, the TrustRRep scheme has a good performance than other schemes.

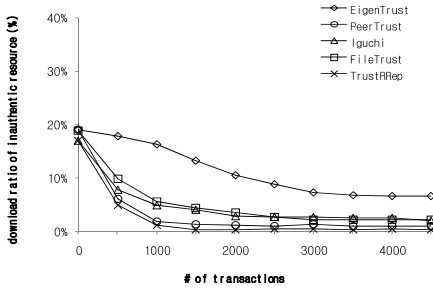


Fig. 3. Case III

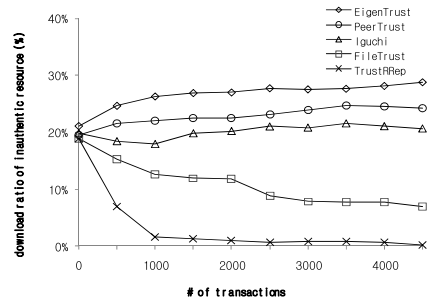


Fig. 4. Case IV

Fig. 3 shows the experimental result of Case III. The EigenTrust scheme shows a low performance as has no consideration for a liar. The FileTrust and Iguchi schemes compute the credibility value referring the feedback provided by peers who download the same resource to identify of a liar. If there is no feedback on the same resource, it is difficult to computation of credibility value. The PeerTrust scheme identifies a malicious peer and a liar using the reputation of peer. The TrustRRep scheme confirms the correctness of feedback generating a random value when a peer submits a feedback and identifies a liar by computing a credibility ratio of peer using equation (1). Fig. 4 shows the experimental result of Case IV. As the EigenTrust scheme has no consideration for a malicious peer and a liar, it has a low performance. The PeerTrust scheme allocates default value that defined in the system to peers'

reputation when peers rejoin. Therefore, because a reputation of peer who rejoins is raised relatively, this scheme has a low performance. The FileTrust scheme has a low performance as the computation time of credibility value is slow. The TrustRRep scheme ignores the download request if the trust value of peer is less than 0. For that reason, the TrustRRep scheme can restrain the participation of a malicious peer.

In the second experiment, we evaluated the effects of change of peers' trust value on the number of downloads and uploads that malicious peers perform in Case IV. Fig. 5 shows the number of in-degrees on malicious peers. The in-degree means the number of download requests. The TrustRRep scheme ignores the download request if the trust value of malicious peers is less than 0. Therefore, the number of in-degrees decreases as transactions are increased. Because the other schemes (EigenTrust, PeerTrust, FileTrust and Iguchi) have no consideration for peers with low reputation or trust value, they have similar performance. Fig. 6 shows the number of out-degrees on malicious peers. The out-degree means the number of uploading resources. The EigenTrust scheme does not identify and restrict a liar. Then the number of out-degrees is nearly same. Because the PeerTrust, Iguchi and FileTrust schemes identify a malicious peer using the reputation of peer, the number of out-degrees diminishes. In the TrustRRep scheme, if malicious peers provide inauthentic resources, its resource reputation is diminished by equation (2). This result affected the computation of peer's reputation by equation (3) and trust value by equation (4). For that reason, the decrement of trust value affects the decrement of the number of out-degrees as transactions are increased. On the other hand, if the malicious peers have high trust value than good peers, they may receive many download requests on resource from peers. Also, they can disseminate the inauthentic resources easily.

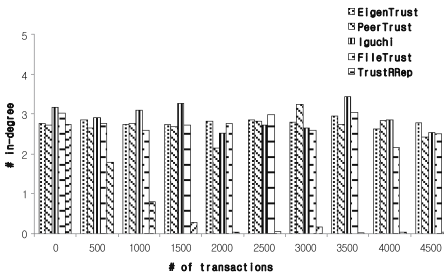


Fig. 5. Number of in-degree

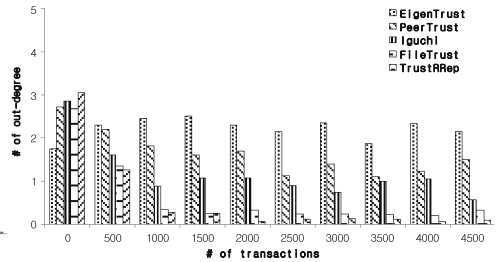


Fig. 6. Number of out-degree

In the third experiment, we observed the number of malicious peers having high trust value than good peers in Case IV. Fig. 7 shows the number of malicious peers having high trust value than good peers having the lowest trust value. Because the EigenTrust and PeerTrust schemes have no consideration on peers' rejoin, the trust value of malicious peers is initialized with default value that defined in the system. Therefore, numerous malicious peers may have high trust value than good peers in the early transaction. The FileTrust scheme refers the feedback of other peers on a resource to compute the credibility value that can identify a liar. If there is no feedback on the same resource, it may occur the computation error of credibility

value. Then the computation error affects the reputation of resource and trust value of peer. On the other hand, the TrustRRep scheme refers the random value generated by equation (5) and confirms the correctness of feedback using equation (6) and (7). Then the TrustRRep scheme identifies a liar by computing the credibility ratio using equation (1). Besides, our scheme does not apply the feedback received from the liar to the computation of resource reputation by equation (2). Therefore, we can improve the correctness on the peer's reputation of equation (3) and the peer's trust value of equation (4) referring the reputation of resource.

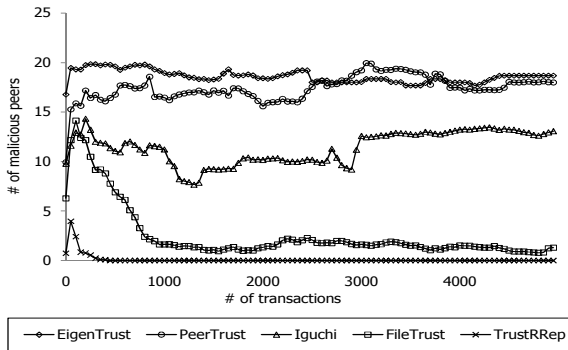


Fig. 7. Computation error of malicious peers

5 Conclusion

A reputation is computed by accumulating the feedback received from all other peers who have interacted with a peer. The usage of reputation in P2P networks can successfully minimize the potential damages to a system and peers can perform a transaction with reliable peers without trusted third parties. Under the deployment of reputation, a peer intentionally plays along with other peers in order to increase/decrease its reputation through dishonest feedback exchanges. In this paper, we proposed the TrustRRep scheme that provide resources and peers with trustworthiness as well as can identify a liar. To evaluate the efficiency of TrustRRep scheme, we have conducted experiments based on the NS-2 simulator. We confirmed that the TrustRRep scheme diminishes the download of inauthentic resources and restricts the participation of malicious peers. Therefore, the TrustRRep scheme can improve the trustworthiness and efficiency of P2P networks by increasing download of authentic resources and by providing a feedback with trustworthiness

Acknowledgments. This work has been supported by research fund from the Korean Land Spatialization Group hosted by the Ministry of Construction and Transportation in Korea.

References

1. Wang, Y., Vassileva, J.: Bayesian Network Trust Model in Peer-to-Peer Networks. In: Moro, G., Sartori, C., Singh, M.P. (eds.) AP2PC 2003. LNCS (LNAI), vol. 2872, pp. 23–34. Springer, Heidelberg (2004)
2. Mekouar, L., Iraqi, Y., Boutaba, R.: Peer-to-Peer's most wanted: Malicious peers. *Computer Networks: The International Journal of Computer and Telecommunications Networking* 50(4), 545–562 (2006)
3. Fall, K., Varadhan, K.: The ns-2 manual. Technical Report, The VINT Project, UC Berkeley, LBL, and Xerox RARC (2002)
4. Kamvar, S.D., Schlosser, M.T., Garcia-Molina, H.: The EigenTrust Algorithm for Reputation Management in P2P Networks. In: Proc. of 12th International Conference on World Wide Web (WWW 2003), Budapest, Hungary, pp. 640–651 (2003)
5. Xiong, L., Liu, L.: PeerTrust: Supporting Reputation-Based Trust for Peer-to-Peer Electronic Communities. *IEEE Transactions on Knowledge and Data Engineering* 16(7), 843–857 (2004)
6. Kwon, O.H., Lee, S.Y., Kim, J.: FileTrust: Reputation Management for Reliable Resource Sharing in Structured Peer-to-Peer Networks. *IEICE transactions on Communications* 4, 826–835 (2007)
7. Iguchi, M., Terada, M., Fujimura, K.: Managing Resource and Servent Reputation in P2P Networks. In: Proc. of 37th Hawaii International Conference on System Sciences (HICSS 2004), Track 7, Hawaii, vol. 7 (January 2004)
8. Jin, Y., Gu, Z., Gu, J., Zhao, H.: A New Reputation-Based Trust Management Mechanism Against False Feedbacks in Peer-to-Peer Systems. In: Benatallah, B., Casati, F., Georgakopoulos, D., Bartolini, C., Sadiq, W., Godart, C. (eds.) WISE 2007. LNCS, vol. 4831, pp. 62–73. Springer, Heidelberg (2007)

Group Key Locking Scheme in Mobile Multicast Environments with Modified MLDs

Byung-Jin Han, Jong-Hyouk Lee, and Tai-Myoung Chung

Internet Management Technology Laboratory,
Electrical and Computer Engineering, Sungkyunkwan University,
440-746, Suwon, Korea
{bjhan,jhlee,tmchung}@imt1.skku.ac.kr

Abstract. In general, when the group key management schemes which have been using in normal multicast environment are applied to mobile multicast environment, they have been experienced some problems because of the mobility of the mobile multicast entities – mobile multicast sender, mobile multicast receiver. We focus on one specific problem. When a mobile multicast sender or receiver moves into a new subnet – called handoff, too many group membership changes occur concurrently. This overwhelmingly many membership changes cause also a lot of group key update requests. The key distribution center is burdened with those requests and does not work properly. The proposed Group Key Locking scheme solves the problem by preventing group key update requests which are caused by handoff of the mobile multicast entities. The results of performance evaluation show that performance of our proposed scheme is better than traditional group key management scheme as 62.5% for mobility of the mobile multicast sender and 14.4% for mobility of the mobile multicast receiver.

1 Introduction

The radical development of wireless network technology and increment demand of multimedia contents and services make mobile multicast an essential technology. Mobile IPv6 [1] supports a mobile node (MN) can change its attachment point even preserving connection. Its coverage spreads not only unicast communication, but also multicast communication. For supporting mobile multicast, Mobile IPv6 introduce two basic methods, known as bi-directional tunneling and remote subscription. To improve performance of mobile multicast, a number of on going researches are progresses in this area [2].

In the mean time, group key management is the most important issue in secure group communication. The group key management schemes do generate group key, distribute group key, disuse group key. If the size of the group is bigger, the cost for group key management is also bigger. So, a scalability is hot issue in group key management [3,4]. A Group key management is also important in mobile multicast. However, in mobile multicast, a concept of mobility is added to multicast. This mobility brings some side effects, especially, breaking scalability.

Those side effects make traditional group key management schemes difficult to use in mobile multicast environment.

In this paper, we propose new group key management scheme which fit in mobile multicast environment. The proposed Group Key Locking scheme introduce new mechanism to improve inadequacy of traditional group key management in mobile multicast.

The rest of this paper is organized as follows. In section 2, we introduce background knowledges and problem statement. Our proposed scheme is described in section 3. In section 4, we evaluate performance of the proposed scheme. Finally we conclude this paper in section 5.

2 Backgrounds

2.1 Group Key Management

Secure multicast aims to preserve authenticity and secrecy for all group communication. It means that, only the registered members of a multicast group are allowed using multicast communication with registered members. However, there is no network-level access control in the Internet. Hence, data encryption is employed to protect the multicast traffic. Here, the data encryption method requires a group key management solution to distribute and maintain cryptographic keys [5,6,8].

Group key management is totally influenced by group membership information. Hence, there are several criteria for examining group key management solution such as scalability, join/leave secrecy, number of keys with a controller, number of keys with each group members, and processing time for key management. The scalability is an ability of scaling very large, wide spread group and handling very frequent key update. Join secrecy means that the new group members cannot read past messages. Leave secrecy means that the past group members cannot read current or future messages [7,8].

There are two relative elements in group key management – key distribution centers (KDC) and an access control list (ACL). A group key management scheme needs an information of the ACL because the group key is a frontend of access control of the multicast distribution tree. Which means that the only allowed group members are permitted to receive multicast traffic. A group key management scheme also needs entity KDC whose main responsibilities are refreshing and distribution of the group key [4,9].

2.2 Mobile Multicast

A multicast member may be a static node or a mobile node. With mobile IP, an user can moves into a new subnet without reconfiguration – handoff. Unfortunately, when a multicast receiver is mobile, it suffer from several problems due to handoff – packet loss, packet duplication, packet out of order, and especially additional delay. The approach that how to combine the multicast and the mobile

IP has an effect on those problems. Mobile IPv6 introduce two basic methods, known as bi-directional tunneling and remote subscription. According to [2,10], bi-directional tunneling has a number of drawbacks such as sub-optimal routing, central point of failure, and overhead of home agent (HA). Thus, in this paper, we employ the remote subscription approach [2].

Similarly, the mobility of the multicast sender causes disruption of multicast session or entire multicast tree reconstruction. The tree reconstruction is most costly work, thus, the other problems are not fully focused. To handle those problems, many solutions are proposed. Supporting Mobile SSM Sources for IPv6 (MSSM)[10] is one of those solution and we take it to provide mobility of sender in mobile multicast environment.

2.3 Problem Statements

In the mobile multicast environment, a multicast sender can have a mobility. When the mobile multicast sender hands off to a new subnet, the multicast receivers must join the new multicast delivery tree $(nCoA, G)$ — the notation (S, G) means that multicast channel which is represented by the multicast source address S and the multicast group address G , so, $(nCoA, G)$ means that a multicast channel whose source address is $nCoA$ and group address is G —. After receiving multicast traffic on the new channel $(nCoA, G)$, the multicast receivers leave the old channel $(oCoA, G)$ [10]. Let the series of above actions call as a channel migration. A handoff of the mobile multicast sender causes massive group membership change and subsequently too many group key update.

This is a very significant problem because processing all the membership changes and the group key updates requires a lot of time delay, and the scalability of group key management is broken by those delay. In the traditional group key management scheme, this problem cannot be solved. So, we propose new group key management scheme to solve this problem.

3 Group Key Locking Scheme

In this section, we describe our proposed group key management scheme which named Group Key Locking scheme. The proposed scheme solves the problem mentioned above in the mobile multicast environment. Now, we provide an overview of Group Key Locking scheme.

3.1 Overview

A handoff of the mobile multicast sender triggers a channel migration. Note that, after the channel migration done, all the members of the old and new channels are the same. So, the group key can be reusable after channel migration. It can be done by making the situation that there is no member who newly joins or leaves the group during channel migration. In this situation, because the group key is reusable after channel migration, processing group key updates during

channel migration only wastes time, resource, and processing power with no gain. Thus, Group Key Locking scheme focus on preventing group key update during channel migration. The Multicast Listener Discovery (MLD) protocol is used to manage group membership such as joining and leaving the group.

To lock the group key and to prevent membership change during channel migration, firstly, distinguishing the MLD messages [11] between a normal MLD message and a MLD message generated by handoff is needed. If the MLD messages are distinguishable, the ACL can ignore MLD messages generated by handoff. So, our proposed GKL scheme assumes the MLD messages generated by handoff does not influence on the ACL. Since the group key update request is based on the information of the ACL, The MLD messages generated by the handoff do not cause the group key update. So, we suggest adding two bits in MLD message to identify its purpose. After distinguish the MLD messages, an ACL only refer normal MLD messages. The mobile multicast sender notify receivers, when it has performed handoff. The KDC just enqueues and does not process the requests of group key update until the mobile multicast sender notify receiver that its handoff done. Because the request of group key update is according to an ACL, all enqueued requests are the normal purpose MLD messages. After unlocking the group key update, the KDC processes enqueued requests in orderly.

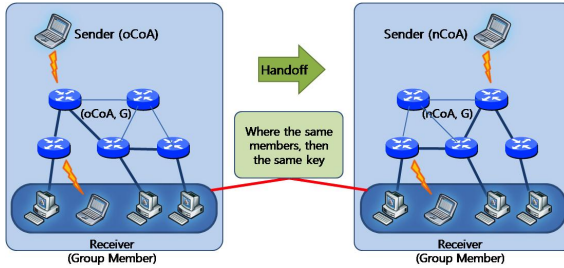


Fig. 1. The main idea of Group Key Locking scheme

Fig.1 describes our proposed GKL scheme by using notation (S, G) . When the mobile multicast sender handoff, its address is changed from $oCoA$ to $nCoA$. Since we make a situation that there is no joining and leaving the group during channel migration, the members of $(oCoA, G)$ and the members of $(nCoA, G)$ become the same. So, a group key of $(oCoA, G)$ can be reusable for $(nCoA, G)$.

So far, we concentrate on the mobile multicast sender perspective. Now, we deal with the mobile multicast receiver perspective. The main idea of the mobile multicast receiver is similar to that of the mobile multicast sender. When a mobile multicast receiver moves into a new subnet, it acquires its new CoA and then, the mobile multicast receiver subsequently joins (S, G) via a new multicast route (nMR). after receiving multicast traffic via the nMR, the mobile multicast receiver leaves (S, G) via a old multicast router (oMR). So, the members of the channel (S, G) are not changed. The more detail procedures are described in later subsection.

3.2 Messages

The proposed GKL scheme needs to modify MLD message for distinguishing between normal MLD messages and handoff generating MLD message, and also needs to introduce new control messages named GKL Source message, GKL Receiver message, GKL Done message, and GKL Ack message.

MLD Modification. The MLD message controls a group membership. The original MLD message does not provide a means of distinguishing itself that whether it was generated by handoff or not. The ability of distinguishing is very important part of GKL scheme. So, we modify the MLD message to be distinguishable by adding two bits in the reserved field.

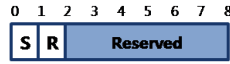


Fig. 2. The modified reserved field

Fig. 2 depict the modified reserved field. We omit the whole field of MLD message for convenience sake. The interpretation of the S bit is that this MLD message is caused by the handoff of the mobile multicast sender. If the mobile multicast sender perceives its handoff and then notifies the receivers of its handoff via the channel $(oCoA, G)$, the receivers generate S bit set MLD messages to join new channel $(nCoA, G)$ and leave old channel $(oCoA, G)$. Similarly, the interpretation of the R bit is that this MLD message is caused by the handoff of the mobile multicast receiver. If the mobile multicast receiver perceives handoff, it generates R bit set MLD messages to join (S, G) via nMR and leave (S, G) via oMR. In normal situation, the S and R bits must not be used.

The S or R bit set MLD messages are treated as special purposed MLD messages. Those messages must not be referred by an ACL in order that the group key update by the KDC would not be influenced by Those messages. Thus, the S and R bit set MLD messages reduce the cost of processing group key update caused by the handoff of the mobile multicast sender or receiver.

New Control Messages. As mentioned earlier, our proposed GKL scheme needs new control messages – GKL Sender message, GKL Receiver message, GKL Done message and GKL Ack message. The details of those messages are shown below.

- *GKL Sender message* : This message is sent to the KDC by the mobile multicast sender when it achieves nCoA in new subnet. Fast handover [12] help to achieving nCoA before the handoff occurs. If the KDC receives this message, it enqueues the requests of group key update for channel $(oCoA, G)$ and channel $(nCoA, G)$, and does not process the enqueued requests until receiving GKL Done message, In other words, the KDC locks the group key of channel $(oCoA, G)$ and channel $(nCoA, G)$. The information in this message consist of

oCoA and nCoA of the mobile multicast sender itself, and the group address G . Set of these three factors are named triple.

- *GKL Receiver message* : This message is sent to the KDC by the mobile multicast receiver who is the member of (S, G) . When the mobile multicast receiver handoff itself, it transmits this message to the KDC. If the KDC receives this message then it enqueues the requests of group key update for channel (S, G) , and does not process enqueued requests until receiving GKL Done message. The contained information in this message consists of oCoA and nCoA of the mobile multicast receiver, a multicast sender address S , and the group address G . Set of these four factors are named quadruple.

- *GKL Done message* : This message can be sent to the KDC by both the mobile multicast sender and receiver. The mobile multicast sender or receiver sends this message when finishing the channel migration or the change MR respectively. This message contains corresponding triple or quadruple information.

- *GKL Ack message* : This message is response message of above three messages. It indicates those messages are received. This message is sent by the KDC and only contains information of success or fail.

3.3 GKL with the Mobile Multicast Sender

The KDC processes the requests of group key update which according to the information of the ACL. Preserving join/leave secrecy is main purpose of group key update. When a mobile multicast sender moves into a new subnet by handoff, its derived effect is very huge. Because, all of the group members subsequently perform the channel migration which consist of joining new channel and leaving old channel, and the channel migration is very costful.

In this section, we describe how GKL scheme works and how it can handle cost and preserve scalability. we employ MSSM [10] for mobile multicast environment. In MSSM, when the multicast source hands off and its address is changed from $oCoA$ to $nCoA$, all the listeners on channel $(oCoA, G)$ join the new channel $(nCoA, G)$ and then leave the old channel $(oCoA, G)$. In MSSM, there exists the problem we pointed out earlier. The procedure to solve the problem is below. In Fig. 3, following procedure is shown.

- Step1*: A mobile multicast sender (with oCoA) configures new CoA when it perceives its handoff by using a protocol such as Fast Handover.
- Step2*: The mobile multicast sender (with oCoA) sends a GKL Sender message to the KDC.
- Step3*: When the KDC receives a GKL Sender message, it locks the group key update for $(oCoA, G)$ and $(nCoA, G)$. And then, KDC sends a GKL Ack message to the mobile multicast sender.
- Step4*: After receiving a GKL Ack message, the mobile multicast sender transmits a Multicast Source Handoff Notification message [10] to receivers via channel $(oCoA, G)$.

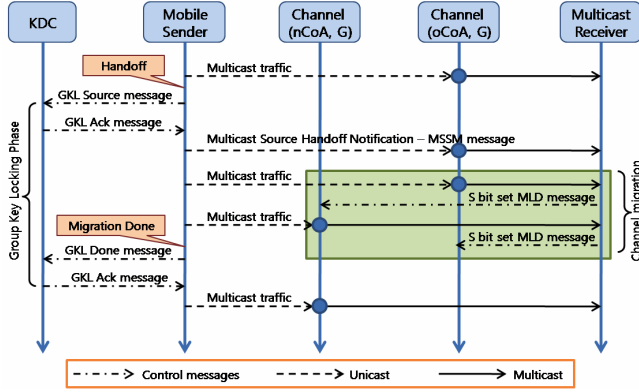


Fig. 3. The procedure of GKL with the mobile multicast sender

- Step5:* All of the receivers migrate channel from $(oCoA, G)$ to $(nCoA, G)$ by using the S bit set MLD messages. Channel migration divides into two sub-steps such as joining and leaving. At the each sub-step, the S bit is set in the MLD message. At the same time, the mobile multicast sender transmits multicast traffic to receivers via both $(oCoA, G)$ and $(nCoA, G)$.
- Step6:* When aware of no more $(oCoA, G)$ member, the mobile multicast sender (with $nCoA$) sends GKL Done message to the KDC.
- Step7:* The KDC unlocks the group key for $(nCoA, G)$ and $(oCoA, G)$. And then, it sends GKL Ack message to the mobile multicast sender. Note that the channel $(oCoA, G)$ and $(nCoA, G)$ has the same members despite of they are not the same channel.

In case of using GKL with the mobile multicast sender, when the mobile source moves into a new subnet, a considerable amount of group key update calculation is reduced by above process. At the *Step5*, receivers uses S bit set MLD messages. By the our assumption, those MLD messages cannot influence on the ACL and cannot generate group key update request also. The KDC is only received group key update requests generated by normal MLD messages during *Step3* to *Step7*. After *Step7*, the KDC processes enqueued requests in orderly.

3.4 GKL with the Mobile Multicast Receiver

We describe the procedure of GKL with the mobile multicast sender in the former section. It is easy to find that handoff of the mobile sender causes huge effect. In contrast with that, in case of the mobile multicast receiver handoff, its effect seems less significant. However, a number of mobile receivers can move into a new subnet concurrently. Its effect can be enlarged.

In this subsection, we describe how GKL scheme resolve the problem mentioned earlier. In Fig. 4, a procedure of GKL with the mobile multicast receiver is shown.

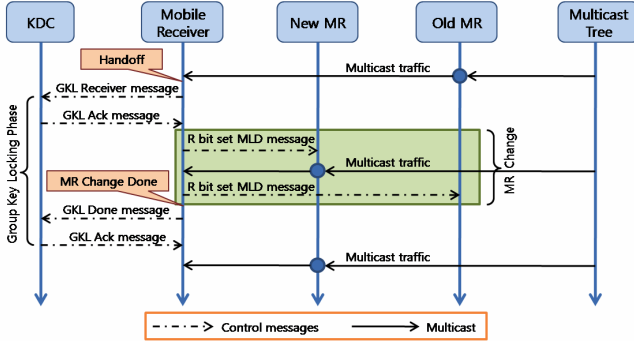


Fig. 4. Procedure of GKL with the mobile multicast receiver

- Step1:* A mobile multicast receiver (with oCoA) configures new CoA when it perceives its handoff by using a protocol such as Fast Handover.
- Step2:* The mobile multicast receiver (with oCoA) sends GKL Receiver message to the KDC.
- Step3:* When the KDC receives GKL Receiver message, it locks the group key update for (S, G) , and sends GKL Ack message to the mobile multicast receiver.
- Step4:* The mobile multicast receiver joins the channel (S, G) via nMR and leaving the channel (S, G) via oMR by using the R bit set MLD messages.
- Step5:* After change multicast router, the mobile multicast receiver sends GKL Done message to the KDC.
- Step6:* The KDC unlocks group key updating for (S, G) and sends GKL Ack message to the mobile multicast receiver.

In case of the GKL scheme with the mobile multicast receiver, when the mobile multicast receiver changes its attachment point, the group key update calculation time for joining group and leaving group is reduced by above procedure. At the *Step4*, receiver uses R bit set MLD messages. By the our assumption, those MLD messages cannot influence on the ACL and can not generate group key update request also. The KDC is only received group key update requests generated by normal MLD messages during *Step3* to *Step6*. After *Step6*, the KDC processes enqueued requests in orderly.

4 Performance Evaluation

In this section, we evaluating performance of our proposed scheme and discuss the result. The proposed GKL scheme has two perspectives such as mobile multicast sender side and the mobile multicast receiver side. The listed parameters in Table 1 are based on [13].

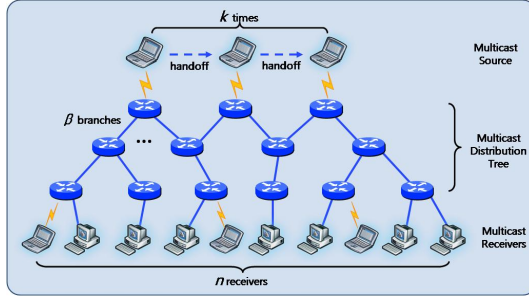
The parameter L_M is the sum of join latency and leave latency. A procedure of group key update consist of key generation and key encrypting. The parameter L_K reflects that fact. The values of parameter β and λ is our assumption and the others are variable.

Table 1. Performance evaluation related parameters

Parameter	Description	Value
L_M	The delay caused by membership change join and leave	100 ms
L_K	The delay caused by group key update procedure	5 ms
β	The number of branch of each multicast router	10
λ	Probability of handoff during lifetime of the group key	0.2
n	The number of the receiver of group G	-
k	The number of handoff of the mobile multicast sender	-
s	The ratio of static nodes	-
m	The ratio of mobile nodes	-

4.1 Evaluation of GKL with the Mobile Multicast Sender

In case of mobile multicast sender perspective of proposed GKL scheme, a performance evaluation model is assumed like as Fig. 5. There are only one mobile multicast sender and n multicast receivers. If the mobile multicast sender moves into a new subnet, its address is changed. This changing causes overwhelmingly huge amount of massive membership changes.

**Fig. 5.** An analysis model for the GKL scheme with the mobile multicast sender

The total delay caused by handoff of the mobile multicast sender D_{sh} is divided by two parts. The one is membership change part and the other is group key update part. Both parts are influenced by the number of group members denoted by n . The total delay of membership change part could do hierarchically and simultaneously. It means that the total delay of membership change part is influenced by the degree of the multicast distribution tree. As shown in Table 1, we assume the number of branch of each multicast router β and this is the degree of the multicast distribution tree. However, the total delay of group key update is not influenced by the degree of the multicast distribution tree and handled linearly by the KDC.

In the case of using traditional group key management scheme, $D_{sh-trad}$ is the sum of both two parts. By contrast, our proposing GKL scheme, D_{sh-GKL} does not contain group key update part because the GKL scheme locks the group key during channel migration.

$$D_{sh-trad} = L_M \cdot \beta \cdot \log_{\beta} n + L_K \cdot n \quad (1)$$

$$D_{sh-GKL} = L_M \cdot \beta \cdot \log_{\beta} n \quad (2)$$

Equation (1) and (2) represent the delay D_{sh} for traditional group key management scheme and our proposed scheme respectively. The comparison result of between equation (1) and (2) is shown in Fig. 6. The result could be more dramatic where n , β or the number of handoff k becomes bigger.

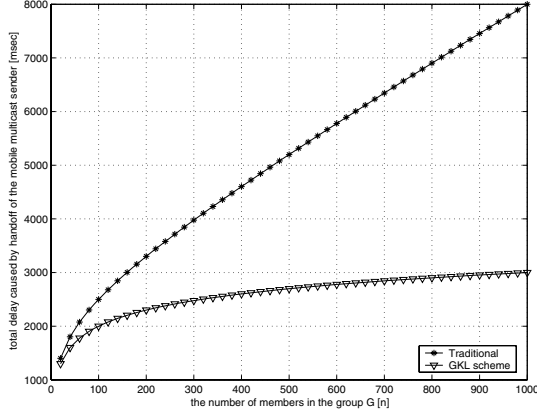


Fig. 6. The delay caused by handoff of the mobile multicast sender

4.2 Evaluation of GKL with the Mobile Multicast Receiver

In contrast to the mobile multicast sender perspective, an analysis model of the GKL scheme with the mobile multicast receiver is depicted as Fig. 7. There are one static multicast sender and n multicast receivers. The probability of handoff of the mobile multicast receiver during lifetime of the group key is given λ . The ratio of static and mobile multicast receiver is s to m .

As the mobile multicast receiver moves into a new subnet, it joins (S, G) via nMR and leaves (S, G) via oMR. This causes group key update. Since several receivers could perform handoff concurrently, we need to calculate N_{avg} , the

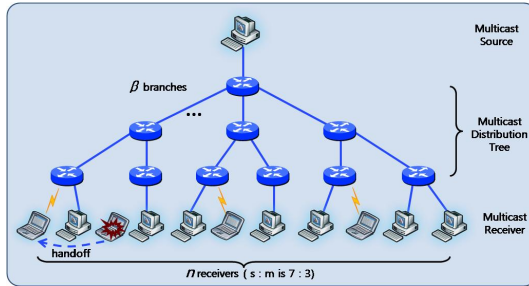


Fig. 7. An analysis model for the GKL scheme with the mobile multicast receiver

average number of receiver who performing handoff during lifetime of the group key. The number of mobile multicast receiver is expressed by $\frac{n \cdot m}{(s+m)}$. Thus N_{avg} can be represented by equation (3).

$$N_{avg} = \frac{\lambda \cdot n \cdot m}{(s+m)} \quad (3)$$

The total delay caused by handoff of the mobile multicast receivers in (S, G) during lifetime of the group key is denoted by D_{rh} . It can be calculated by similar manner to the case of the mobile multicast sender.

$$D_{rh-trad} = L_M \cdot \beta \cdot \log_{\beta} N_{avg} + L_K \cdot N_{avg} \quad (4)$$

$$D_{rh-GKL} = L_M \cdot \beta \cdot \log_{\beta} N_{avg} \quad (5)$$

Equation (4) and (5) represent the total delay D_{rh} of the traditional group key management scheme and our proposed GKL scheme respectively. In Fig. 8, the comparison result between equation (4) and (5) is shown. The gap of two lines would be wider where n , β , λ or $\frac{m}{(s+m)}$ becomes bigger.

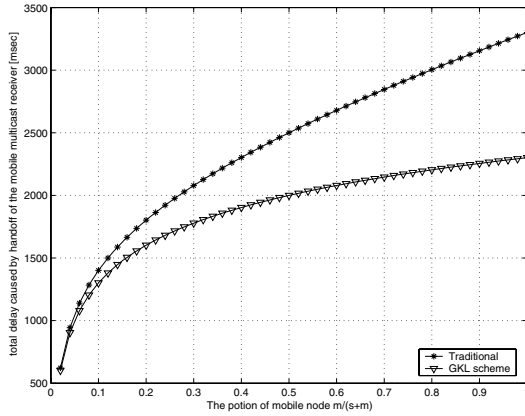


Fig. 8. The delay caused by handoff of the mobile multicast receiver

4.3 Result Analysis

Using parameters in Table 1 and above equations, we compute the performance enhancement. Equation (6) and (7) represent the performance enhancement of our proposed GKL scheme.

$$Enhancement_{sender} = (1 - \frac{D_{sh-GKL}}{D_{sh-trad}}) \times 100\% \quad (6)$$

$$Enhancement_{receiver} = (1 - \frac{D_{rh-GKL}}{D_{rh-trad}}) \times 100\% \quad (7)$$

When a situation as $n = 1000$, $\beta = 100$, $\lambda = 0.2$, and $\frac{m}{(s+m)} = 0.3$, the computed $Enhancement_{sender}$ is 62.5% and the computed $Enhancement_{receiver}$

is 14.4%. Those results clarify that Our proposed GKL scheme is very efficient in mobile multicast environment and especially, for the GKL scheme with the mobile multicast sender case.

5 Conclusions

In this paper, we propose Group Key Locking scheme in mobile multicast environment. The traditional group key management schemes does not fit in mobile multicast environment, because these has no ability to handle mobility of the mobile multicast entities. To solve this problem, we proposed Group Key Locking scheme. Our proposed scheme reduces cost of group key update by preventing request of group key update from handoff generating MLD messages. We calculate the performance enhancement of the proposed scheme. As a result of comparison between traditional group key management scheme and our proposed GKL scheme, shows that 62.5% enhancement for the mobility of the multicast sender and 14.4% enhancement for the mobility of the multicast receiver. In future work, we merge our scheme to recognized scheme in traditional group key management area.

Acknowledgement

This research has been supported by a grant of the Korea Health 21 R&D Project, Ministry of Health & Welfare, Republic of Korea (A020608).

References

1. Johnson, D., Perkins, C., Arkko, J.: Mobility Support in IPv6., RFC 3775 (June 2004)
2. Romdhani, I., Kellil, M., Lach, H.-Y.: IP Mobile Multicast: Channlenges and Solutions. *IEEE Communications Surveys & Tutorials*, 18–41 (First Quarter, 2004)
3. Sun, Y., Liu, K.J.R.: Scalable Hierarchical Access Control in Secure Group Communications. In: *IEEE INFOCOM 2004*, pp. 1296–1306 (March 2004)
4. Ballardie, A.: Scalable Multicast Key Distribution, RFC 1949 (May 1996)
5. Kellil, M., Romdhani, I., Lach, H.-Y.: Multicast Receiver and Sender Access Control and its Applicability to Mobile IP Environments: A Survey. *IEEE Communications Surveys & Tutorials*, 46–70 (Second Quarter, 2005)
6. Adusumilli, P., Zou, X., Ramamurthy, B.: DGKD: Distributed Group Key Distribution with Authentication Capability. In: *IEEE SMC Information Assurance Workshop 2005*, pp. 286–293 (June 2005)
7. Rafaeli, S., Hutchison, D.: A Survey of Key Management for Secure Group Communication. *ACM Computing Surveys*, 309–329 (September 2003)
8. Moyer, M., Rao, J.R., Rohatgi, P.: A Survey of Security Issues in Multicast Communications. *IEEE Network*, 12–23 (November-December 1999)
9. Wallner, D., Harder, E., Agee, R.: Key Management for Multicast: Issues and Architectures. RFC 2627 (June 1999)

10. Jelger, C.S., Noel, T.: Supporting Mobile SSM Sources for IPv6. In: IEEE GLOBE-COM 2002, pp. 1693–1697 (November 2002)
11. Vida, R., Costa, L.: Multicast Listener Discovery Version 2 (MLDv2) for IPv6, RFC 3810 (June 2004)
12. Koodli, R.: Fast Handovers for Mobile IPv6, RFC 4068 (July 2005)
13. Setia, S., Koussih, S., Jajodia, S., Harder, E.: Kronos: a scalable group re-keying approach for secure multicast. In: IEEE Symposium S&P 2000, pp. 215–228 (May 2000)

The Best Practice and Evaluation of Processes to Develop OSSs Using XML-Based Interfaces

Yuki Kishikawa, Daisuke Fujishima, Hironao Tamura,
Kazuhide Takahashi, and Shoichi Hirata

Network Development Department
NTT DoCoMo, Inc.

3-5 Hikari-no-oka, Yokosuka-shi, Kanagawa, 239-8536, Japan

Tel.: +81-468-40-3773; Fax: +81-468-40-3784

{kishikawa, fujishimad, tamurahir, takahashikazu,
hiratas}@nttdocomo.co.jp

Abstract. We propose the best practice of improving OSS development processes by modifying the conventional waterfall model. This method is based on the interfaces between OSSs and NEs expressed by XML, which enables us to reduce the cost of developing OSSs and shorten their development period even if new specifications are added. For further efficiency, users themselves could implement their own specifications. This could be regarded as a kind of Software as a Service (SaaS) that has lately been attracting more attention in many fields. The proportion of the scale to develop OSSs using XML to the whole scale was consequently 80-90%, and the development scale of XML-coding at a certain periodical development was 5% of the entire development scale of XML-coding in this system. This shows that we succeeded in drastically improving the efficiency of developing OSSs.

Keywords: OSS, XML, SaaS, Best practice.

1 Introduction

Core network systems have recently been actively standardized to enable progression toward all-IP networks by using All IP Based Networks (AIPNs) [2] or Next Generation Networks) (NGNs) [3] that offer highly convenient communication services to general consumers, which have been studied by the 3rd Generation Partnership Project (3GPP) or the International Telecommunication Union-Telecommunication (ITU-T). As a result, it is anticipated that the enhancements or migration to such progressive networks will be rapidly promoted. Along with these changes, numerous new NEs that have various interfaces with Operations Support Systems (OSSs) are expected to be introduced within these networks. Here, we have merely called the interface specifications between OSSs and NEs “NE external specifications” or “NE specifications”.

The OSSs play a crucial role in making the work, such as the enhancement or migration of networks and NEs, more efficient. This is why OSSs are indispensable systems for mobile network operators to acquire market share. Therefore, the main

issue is to inexpensively develop the application software for OSSs to maintain NEs that have new external specifications, and to promptly and appropriately meet the demands for addition functions by OSS users.

To resolve this issue, we have developed large-scale OSSs by using Intel Architecture (IA) servers for hardware and open source software for OS/middleware. As a result, we succeeded in substantially cutting the Total Costs of Ownership (TCO). The applications of OSSs are divided into logical elements (LEs), which are modularized in each parallel processing unit, and these LEs are separately installed in thousands of IA servers. We called this the “Distributed Data Driven Architecture (D3A) [4]”. Fig. 1 is a schematic of the D3A. In this architecture, we made it a rule to use Java code taking into consideration the portability of its applications, and we also used a great deal of XML to enable more efficient development. We conformed to a traditional waterfall process when developing D3A using XML. However, it was difficult in practice in this process to conform to the required specifications in the early phases, and it also became problematic to change the external specifications or design as development progressed [5] [6].

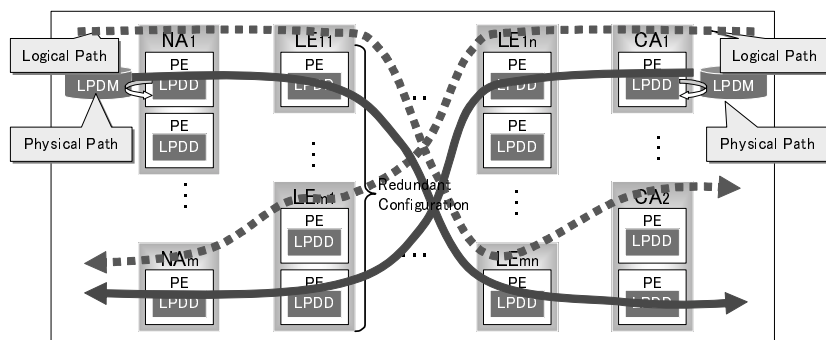


Fig. 1. Schematic of Distributed Data Driven Architecture (D3A)

In this report, we have classified external OSSs into “NE interface specifications” and “user interface specifications”. We called the former “NE specifications” as previously mentioned, and the latter “user specifications”. First, we propose a way of quickly and inexpensively developing applications of OSSs to which new NE specifications are added by expressing them in XML. We also discuss development processes suitable for XML obtained by modifying the conventional waterfall model. Second, we introduce a way for users themselves to be able to implement the user specifications by disclosing the Application Programming Interface (API) of OSS applications through our internal corporate network. Finally, we evaluate quantitatively the development efficiency of our large-scale OSS shown above.

2 Approaches

The specifications are different in each NE, and especially in each of its venders, and the required conditions are different for all user specifications, especially window specifications. Fig. 2 shows these time-series behaviors. According to the time-series

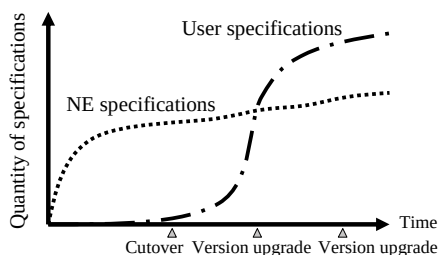


Fig. 2. Time evolution of OSS specifications

behaviors of the “required” specifications in this figure, the issue with NE specifications is how efficiently we can implement them “in the middle of development”, and the issue with user specifications is how efficiently we can do this “after they have been released. Furthermore, implementing all required conditions involves huge development costs for user specifications (especially window user friendliness) as users have strong preferences, and their individual requirements differ greatly.

First, let us discuss how to implement NE specifications efficiently “in the middle of development”. Traditional OSS applications are often developed “partially” in “heavyweight programming languages” such as C++ or Java, which need compilations. We expect that the productivity of OSS can be drastically improved if we prepare frameworks for OSS applications and express NE specifications as data and scripts by using these “lightweight programming languages” which compilations are unnecessary. XML is a “lightweight programming language” because it can not only be used free of charge but it is also a generalized markup language that can express various data structures. Moreover, as performance by IA servers has been significantly improved in recent years, XML has become more freely available.

Second, we will discuss the development process using XML. The Ajail development [7], which rectified the fault with the waterfall model previously mentioned, has recently attracted some attention. However, this development process is often applied to “small-scale” systems used by about ten people. In our “large-scale” OSSs, the development process using XML was based on the “modified” waterfall model. Fig. 3 outlines this process. It consists of ten development processes. Appropriate feedback of a defect correction and a reinforced testing is given to the previous process. In our large-scale OSSs, the efficiency of creating design documents related to XML was low because their creation was decentralized in each design process. Refer to our report [4] for the details on the process to develop D3A.

To improve the efficiency of developing OSS applications, we focused on easy-to-understand features of XML that were in text form with tags. In other words, we expected more content could be omitted from development documents than that in the conventional development of OSS applications where compilation languages are mainly used, since programmers could easily understand the meaning of XML source codes. Furthermore, since we could review the design documents and codes more easily, more defects could be extracted in each design process. Conversely, since it was possible to maintain the highest quality and to satisfy the performance requirements in each design process, we could cut down on the number of development man-hours in all testing processes. Consequently, we decided to modify the waterfall model so that it would suit the development process using XML.

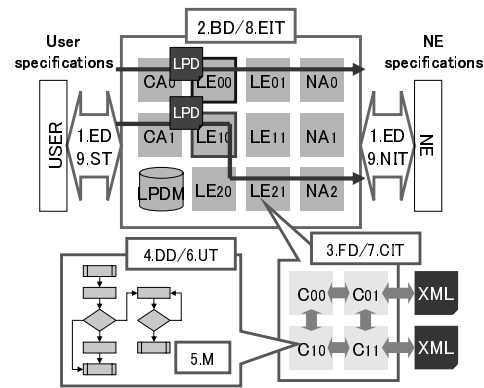


Fig. 3. Outline of conventional development process based on waterfall model

Finally, we will discuss how efficiently we can implement user specifications “after they have been released”. SaaS is a service model that has attracted a great deal of attention due to the rapid growth of Salesforce.com [8], which offers the on-demand Customer Relationship Management (CRM). We focused on the feature of SaaS that “users could customize applications by using APIs”. That is, we expected that OSSs could be developed more efficiently and their development period shortened, if we offered APIs to OSS users and let them customize their own user specifications by using lightweight programming languages, such as XML and JavaScript. Even though this model only offers application services to OSS users through corporate networks, we considered it included the concept of SaaS in a broad sense.

3 Expressing NE Specifications by XML

This section examines OSS mechanisms where alterations and additions to NE specifications can be changed flexibly. The periodical addition of functions to existing NEs, and the addition of functions to OSSs when new NEs are introduced can be classified into eight patterns of external specification changes or additions: alarm specifications, command specifications, equipment composition, NE- and OSS-composition data specifications, traffic-data-item specifications, work procedures, and congestion-control operation procedures. Here, we will only discuss the mechanism for flexibly implementing the alterations and additions to NE specifications in two cases of NE- and OSS-composition data specifications and work procedures due to limitations of space.

3.1 Alterations and Additions to NE- and OSS-Composition Data Specifications

The specifications for NE- and OSS-composition data are different in all NEs. The physical structure of NE-composition data takes various forms such as binary files, text format files, and files of command forms, and their specifications differ. The logical meaning of NE data and the conditions for checking the consistency of NE-composition data are also complicated and extensive. The OSS-composition data

items are automatically extracted from the NE-composition data and data items by user inputs.

We described all the NE- and OSS-composition data structures by using XML to improve the efficiency of development, and decided to permit OSS users to edit these themselves using XML editors. This meant that it was unnecessary for us to develop the windows for editing them. Moreover, the conditions for checking the consistency of the value types or ranges were defined with the style sheets of the XML editors. As a result, we could cut down on the cost of developing Graphical User Interfaces (GUIs) to input both data that makes up most of the cost in conventional systems. Users could also detect mistakes in their input while inputting (e.g., these were displayed in red). The conditions for checking consistency in data items or tables with both data could be described by using XML. Furthermore, we could describe the logic for converting the previously mentioned forms by using XML. By using these processes, the alterations and additions to both data specifications could be flexibly implemented by using mere descriptions of XML. Refer to our report [9] for the details on these processing methods.

3.2 Alterations to Work Procedures

There are various work contents to operate NEs, such as file upgrades, patches for plug-ins (for bug fixes), and alterations to NE-composition data. These procedures are occasionally changed for the purpose of improving the work efficiency. They are also changed along with alterations or additions to alarm and command specifications. It is also necessary to take the conditions OSS users require for their work procedures into consideration.

We made it possible to describe the work procedures utilizing XML scripts by using D3A with a dimensional structure (D4A: Dimensional D3A). Refer to our report [10] for the details on the ways the work procedures were processed. As will be described later, these XML-script specifications and APIs are disclosed to OSS users, so they can design and program external specifications by themselves for the work procedures they need. Therefore, it is possible to flexibly cope with alterations and additions to work procedures, alarm specifications, and command specifications.

Also, NE specifications can easily be expressed by using XML for the other 6 patterns. We can also easily implement additional functions by combining these patterns. There are very few scripts that need to be coded in Java.

4 Evaluation of Development Processes Suitable for XML

We will now discuss the elimination and consolidation of development processes as to the eight patterns mentioned above from the features of D3A/D4A that NE specifications can be expressed by using XML. Table 1 lists the improved development processes for all NE specifications. This consists of seven kinds of development processes as follows: ED, XD, M, XT, EIT, NIT, and ST.

Table 1. Improved XML development processes

Existing type of NE	1.ED	2.BD	3.FD	4.DD	5.M	6.CT	7.CIT	8.EIT	9.NIT	10.ST
(1) Protocol	ED	XD			M				NIT	
(2) Alarms	ED	XD			M				NIT	
(3) Commands	ED	XD			M				NIT	
(4) Equipments configuration	ED	XD			M				NIT	
(5) NE/OSS configuration data	ED	XD			M	XT			NIT	ST
(6) Traffic data	ED	XD			M			XD		ST
(7) Works procedures	ED	XD			M	XT			NIT	
(8) Congestion control procedures	ED	XD			M	XT			NIT	ST

New type of NE	1.ED	2.BD	3.FD	4.DD	5.M	6.CT	7.CIT	8.EIT	9.NIT	10.ST
(1) Protocol	ED	BD	FD	DD	M	UT	CIT	EIT	NIT	ST
(2) Alarms	ED	XD			M			EIT	NIT	ST
(3) Commands	ED	XD			M			EIT	NIT	ST
(4) Equipments configuration	ED	XD			M			EIT	NIT	ST
(5) NE/OSS configuration data	ED	XD			M	XT		EIT	NIT	ST
(6) Traffic data	ED	XD			M			XT		ST
(7) Works procedures	ED	XD			M	XT		EIT	NIT	ST
(8) Congestion control procedures	ED	XD			M	XT		EIT	NIT	ST

The NE specifications are reviewed in the ED, and ambiguities and contradictions in the NE specifications are excluded. Since the XML source code is the OSS internal expression of NE specifications, refinements to internal structures are unnecessary in the design process. Therefore, we integrated conventional BD, FD, and DD into one process, and we only created the XML design documents in this process by converting NE specifications into internal expression forms. We called this process XD. Consistency between XML specification and NE specifications was verified by reviewing them in XD, and whether sufficient defects mixed in during the XML design process had been removed was evaluated. The XT items for the testing were extracted in this process at the same time. The XML source coding in XM conformed to XML-design specifications. Testing processes corresponding to XD were called XT. This development process integrated conventional UT and CIT. The appropriateness of XML source codes was verified in XT, and whether sufficient defects mixed in during the XML-coding process had been removed was evaluated. Compatibility with NE specifications was verified in NIT, and whether residual defects mixed in during the XML-design and XML-coding processes had been sufficiently removed was evaluated. Performance and long-term stability were also evaluated in ST.

5 Application of SaaS to User Specifications

User specifications can be classified into work procedures and window specifications. These functions are specifications that are required for all OSSs in common, and do not rely on individual NE specifications or individual characteristics of user preferences. In the D3A system, these functions correspond to LEs. Coding with Java is necessary to add new functions to existing LEs or to create new LEs. Therefore, it is not easy for OSS users to customize LEs by only applying SaaS, that is, by making use of disclosed APIs. Therefore, we considered applying SaaS to work procedures and window specifications.

5.1 Application of SaaS to Work Procedures

The XML scripts that stand for work procedures are created by OSS users with client PCs at their offices. OSS users upload the XML scripts they edited from client PCs to

a certain LE via the internal corporate network, and they can complete their work by only setting the NEs as objects for execution and schedules, which are the starting conditions for XML scripts through GUIs. Other users are also able to download the past XML scripts and to remake them for other work procedures. These XML scripts consist of main scripts and component scripts. The calling order of each component script is described in the main script. We can describe the following information to the component script: command input processing, reception processing of command responses, parameter-transfer processing between the main script and component script, a function for arithmetic or character-string operations, control structures (e.g., condition branching and repetition), and file operations. OSS users can flexibly set up their own unique work procedures by combining these APIs. Fig. 4 shows an example of coding XML script.

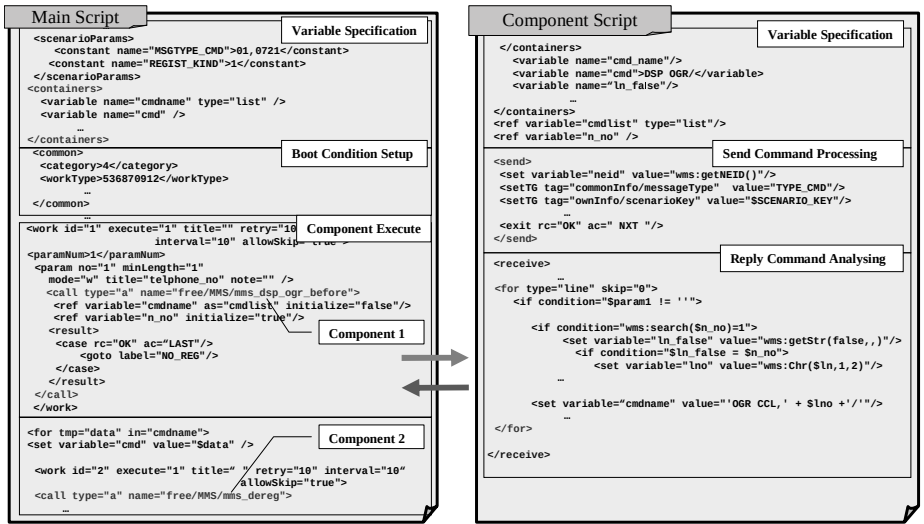


Fig. 4. Example of coding XML script of work procedures

5.2 Application of SaaS to Window Specifications

The source codes that implement the window specifications in our large-scale OSSs, such as JSP, JavaScripts, and stylesheets, are disclosed to OSS users as samples of how they should code. Here, we have called them “sample source codes”. Of course, these sample source codes are originally described by OSS developers and offered in commercial environments. In addition, the Java servlets are open to OSS users as APIs. Therefore, OSS users can modify sample source codes according to their individual purposes or preferences, and they can also create new windows by combining window parts and Java servlets using JSP, JavaScripts, and HTML. Archiving source codes that OSS users have edited and the Java servlets of APIs makes one Web application. By uploading this Web application from client PCs to Client Adapters (CAs) via the internal corporate network, OSS users can utilize new

windows. Here, CA is one of the LEs that carry a Web server. OSS users can open windows they made or original windows by changing URLs.

There are some sample windows customized by OSS users in Fig. 5, which shows the arrangement of various buttons to fix the command parameters; by pushing them properly, OSS users can automatically input command character strings.

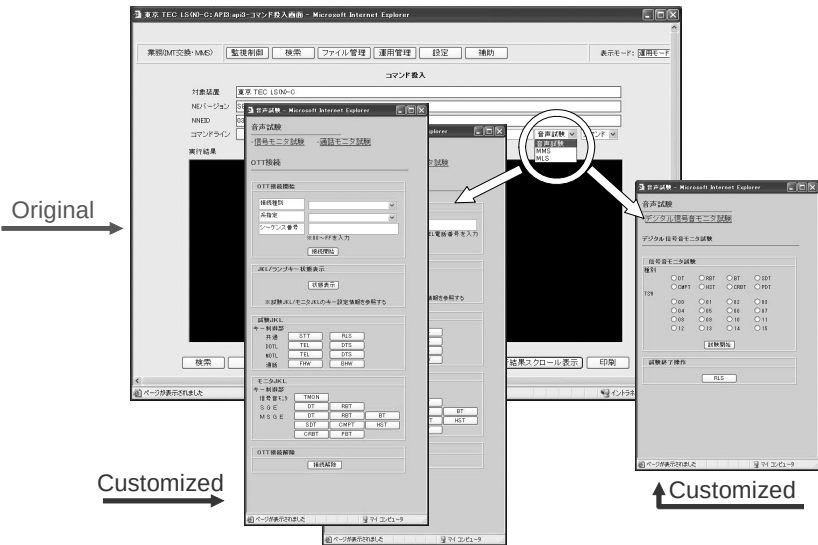


Fig. 5. Sample windows customized by OSS users

6 Evaluation

6.1 Evaluation of Development Scale

We continue to periodically add functions to our OSSs even though the initial development has finished. These periodical updates involve adding new functions to existing LEs, adding specifications or changes to existing LEs, and adding new NE specifications. The time evolution of the proportion of the development scale of XML-coding to the whole scale including Java is shown in Fig. 6. The rate of the coding scale by XML to the total scale is stable at about 60% in initial development, at about 80-90% in the periodical development of addition functions, which means that the mere coding of XML scripts enables many functions to be efficiently added later.

The proportion of development scale of coding in each Java and XML to the whole scale in a certain periodical development of addition functions is shown in Fig. 7. The amount of Java-coding gradually decreased in proportion to the number of file versions, and eventually reduced to about 1% with the fifth version. In contrast, the amount of XML-coding increased and reached 5%. The reason that there was a large amount of Java-coding in the first version is because functions had to be added to existing LEs to compensate for insufficient functions, just after the initial crossover from existing OSSs to new ones. The reason for the large amount of XML-coding

with the second version was because we added functions to cope with two versions of NE specifications. The reason that this percentage fell to about 50% in the third file versions is because many new functions were added to LEs. As previously mentioned, many functions can be achieved by coding with XML in these large-scale OSSs.

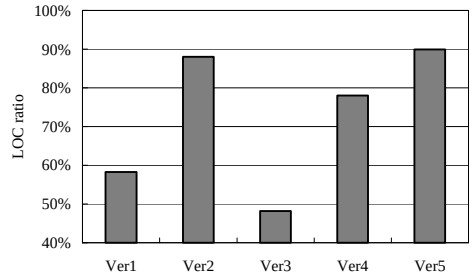


Fig. 6. Ratio between Lines Of Codes (LOCs) coded in XML and Java

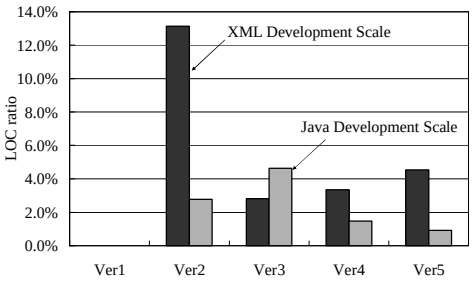


Fig. 7. Ratio between original LOCs and periodically added LOCs

As to work procedures, the ratio of scale of coding XML scripts made by OSS users reached over 60%. Surprisingly, OSS users created about 1800 kinds of work procedures. From this, we found that OSS users were completely accustomed to adding or changing work procedures by themselves. OSS users also seemed to be accustomed to customizing windows specifications as they customized about 20% of the entire windows specifications.

6.2 Considerations

First, let us discuss additions or changes to NE specifications. As previously discussed, these have almost no influence on the development of Java applications. In periodically adding functions to our large-scale OSSs, we can develop their applications by mostly modifying XML scripts even if there are additions to NE specifications according to the evolution of IP-based networks. Furthermore, we improved the conventional waterfall model that has ten development processes into our new scheme that consisted of seven processes, which made the development of OSSs more efficient.

Next, let us discuss additions or changes to user specifications. Although the OSS functions had been given unilaterally to OSS users about existing OSSs, the users themselves became able to make their own jobs more efficient in these large-scale OSSs, which motivated them to improve their work. This was one determining factor in the success of the SaaS strategy. Actually, XML scripts are greatly utilized by OSS users in constructing all-IP networks, and contribute to reducing the number of operations and preventing mistakes. In addition, there are few OSS users who still use the original samples with window specifications. From this, we can see the high operability of the windows that they created, and confirm that another factor in the success of our SaaS strategy lies in its “lightweight programming languages”, that is, programming languages such as XML and JavaScript that OSS users can easily code.

7 Conclusions

We proposed the best practice of improving OSS development processes and quantitatively appraised the efficiency of developing our large-scale OSSs by this method. As a result, the proportion of the scale to develop OSSs using XML to the whole scale was about 80-90%, and the development scale of XML-coding at a certain periodical development of addition functions was about 5% of the entire development scale of XML-coding in this system. This demonstrated that we succeeded in drastically improving the efficiency of developing OSSs.

References

1. SaaS, <http://www.saas.com/>, <http://www.saasshowplace.com/>
2. AIPN, <http://www.3gpp.org/ftp/Specs/html-info/22978.htm>
3. NGN, <http://www.itu.int/ITU-T/ngn/>
4. Takahashi, K., Kon, T., Akiyama, K., Jinguji, M.: The performance evaluation of a Distributed Data Driven Architecture and the Application Design Method. IEICE Trans. Commun. J88-B(7), 1202–1212 (2005) (Japanese edn.)
5. Brooks Jr., F.P.: The Mythical Man-Month: Essays on Software Engineering. Addison-Wesley, Reading (1995)
6. Pete, M.: Software Craftsmanship. Addison-Wesley Professional, Reading (published, 2001)
7. Agile Alliance, <http://www.agilealliance.org/>
8. Salesforce.com, <http://www.salesforce.com/>
9. Kon, T., Tanabe, K., Akiyama, K., Takahashi, K., Jinguji, M.: Creation of Configuration Data Management System Based on a Distributed Data Driven Architecture and XML. In: APNOMS 2005, Poster Sessions P2.4 (September 2005)
10. Tanabe, K., Kon, T., Akiyama, K., Takahashi, K., Jinguji, M.: The realization of Work Management System based on 2-Dimensional Distributed Data Driven Architecture. In: Proceedings of APNOMS 2005, pp. 529–540 (September 2005)

Review of Knowledge Engineering Requirements for Semantic Reasoning in Autonomic Networks

John Strassner¹, Mícheál Ó Foghlú¹, Willie Donnelly¹,
Joan Serrat², and Nazim Agoulmine³

¹Telecommunications Software and Systems Group, Waterford, Ireland
{jstrassner,vdmeer,mofoghlú,wdonnelly}@tssg.org

²Universitat Politècnica de Catalunya, Barcelona, Spain
serrat@tsc.upc.edu

³University of Evry Val d'Essonne, Evry, France
nazim.agoulmine@iup.univ-evry.fr

Abstract. Modern day services use heterogeneous devices and functionality across a set of different types of networks. This mix of device- and vendor-specific technologies makes it difficult to share and reuse what should be common management data. Worse, the focus on technology has ignored how to enable the business to drive network services and resources offered at any given time, as well as how to adapt services and resources offered by the network to changing user needs, environmental conditions, and business objectives. This paper describes the requirements for knowledge engineering to enable semantic reasoning to augment traditional network management to solve the above problems, and describes a new approach as the basis of future work in this area.

Keywords: Autonomic network, data model, information model, knowledge engineering, knowledge representation, ontology, semantics.

1 Introduction

Currently, network management data, as specified and produced by network equipment vendors, is typically defined using data structures conformant to the Structure of Management Information (SMI) [24] using Simple Network Management Protocol (SNMP) applications [1] or custom Command Line Interface (CLI) [2] languages. Usually, vendor- and device-specific data models are used to define management data. There are two problems with this approach. First, there is no standard way to represent *business* concepts, such as a customer or a Service Level Agreement (SLA) using Management Information Bases (MIBs) or CLI. Second, it is difficult, if not impossible, to represent important *semantics* of the management data, since neither MIBs nor CLI include logic programming constructs.

These are symptoms of two larger problems. First, the disciplines of Network Management and Software Engineering are separated. For example, SNMP relies on an informal and incomplete data model, which is manifested by the inability to specify a common way to structure and implement a Management Information Base

(MIB). Second, the disciplines of software engineering and knowledge engineering are also firmly separated. This leads to a fundamental limitation in how network management data can be used. This paper argues that these three disciplines need to be united, as separating them prevents software processes that are used to generate code to control system behavior from taking advantage of knowledge engineering processes that are used to understand data and make decisions.

The basic objectives of autonomic systems [3] [4] [5] *mandate* this combination because of two fundamental reasons. First, the control loop used in autonomic systems requires machine-based learning and reasoning processes to help tune its actions. Second, autonomic systems seek to lessen the dependence on requiring human operators to change system functionality; this means that autonomic systems need to be able to control and change functionality dynamically without (or at least with minimal) aid of a human.

This paper proposes a novel knowledge engineering approach that can take advantage of Unified Modelling Language (UML)-based software engineering and AI-based semantic reasoning to overcome the inherent diversity in vendor-, device-, and technology-specific information by using information models to represent the static characteristics of modelled objects as facts, and ontologies to augment those facts with semantic information so that the facts may be reasoned about. The organisation of this paper is as follows. Section 2 provides a brief introduction to autonomic computing. Section 3 summarizes the advantages of using information and data models, and section 4 summarizes the advantages of using ontologies. Section 5 describes how these two different approaches, each with their own data structures, are combined, and illustrates the approach using a simple but powerful case study. Section 6 summarizes initial experiments, and section 7 summarizes the paper.

2 Goals of Autonomic Computing and Networking

The purpose of autonomic computing is to manage complexity. The name was chosen to reflect the function of the autonomic nervous system in the human body. By transferring more manual functions to involuntary control, additional resources (human and otherwise) are made available to manage higher-level processes.

The fundamental management element of an autonomic computing architecture is a control loop, as defined in [3] [5] [6]. The idea is to instrument a Managed Resource so that an Autonomic Manager can communicate with it in a standardised fashion. Sensors retrieve data, which is then analysed to determine if any correction to the managed resource(s) being monitored is needed (e.g., to correct “non-optimal”, “failed” or “error” states). If so, then those corrections are planned, and appropriate actions are executed using effectors that translate commands back to a form that the managed resource(s) can understand. If the autonomic network can perform manual, time-consuming tasks (such as configuration management) on behalf of the network administrator, then that will free the system and the administrator to work together to perform higher-level cognitive functions, such as planning and network optimization.

Figure 1 shows a simplified version of our FOCAL autonomic architecture [6]. Multiple networks and network technologies require multiple control planes that can use completely different mechanisms; this makes managing an end-to-end service

difficult, since different management mechanisms must be coordinated. FOCAL addresses this through model-based translation, which uses a combination of models and ontologies to translate disparate sensed data into a common networking *lingua franca*. This enables facts extracted from sensor input data to be reasoned about using ontology-based inferencing. More importantly, business goals and objectives can be directly related to the system using context-aware policy models [7] [8] that use the Policy Continuum [9] to relate the needs of different constituencies (e.g., business, network, and programming people) to each other.

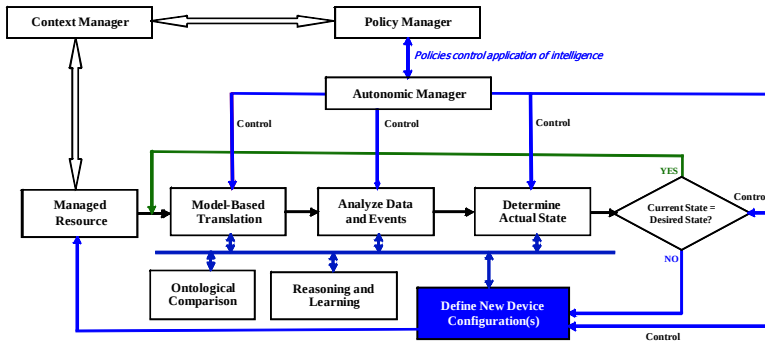


Fig. 1. Simplified Version of the FOCAL Autonomic Architecture

The key to the FOCAL *adaptive control loops* is the interaction between the context manager, policy manager, and autonomic manager. Conceptually, the context manager detects changes in the network, or in user needs, or even in the business; these context changes in turn activate an associated set of policies that define the functionality the autonomic manager should govern. This reconfigures one or more devices, so that the services and resources provided by the autonomic system can *adapt* to these new needs. The autonomic manager uses these policies to govern each of the architectural components of the control loop, enabling the different control loop components to change the type of algorithm used, the type of function used, and even the type of data used as a function of context.

A system built in accordance with FOCAL is self-governing, in that the system senses changes in itself and its environment, and determines the effect of the changes on the currently active set of business policies. In general, those changes could either cause a new set of business policies to be activated, or endanger one or more goals of the currently active set of business policies. In the latter case, FOCAL reconfigures the system to ensure that the currently active set of business policies is not violated and observes the results.

3 Strengths and Weaknesses of Using Information Models

The main advantage of using UML to represent network management knowledge is that there is a wide body of UML and UML-like models already available that are supported by numerous tools. Our work uses the DEN-ng information model, which

is currently being standardized in the Autonomic Communication Forum (ACF) [10]. There are three reasons for this:

1. DEN-ng uses classification theory [11] to distinguish information about class instances from information about how classes and other objects are classified.
2. DEN-ng employs multiple types of abstractions, such as patterns [12]; examples of these include capabilities (normalized functionality independent of platform, device, and vendor) and roles (functionality that a managed entity can take on).
3. DEN-ng uses finite state machines to orchestrate the behavior of a managed object. This goes far beyond other models, which typically are limited to representing the current state of an object; rather, DEN-ng models the *lifecycle* of managed objects in terms of states, and orchestrates their behavior by using context-aware policies to determine which states a managed object is allowed to transition to [7].

The DEN-ng information model defines two broad classes of managed objects – those that are identifiable and those that only have a value. [13] says:

“Many objects are not fundamentally defined by their attributes, but by a thread of continuity and identity... Many objects have no conceptual identity. These objects describe some characteristic of a thing.”

This enables software to realize whether there are hidden semantics behind a managed object. For example, an entity that has an associated identity has, by definition, a set of additional semantics that further represent its behavior and interaction with other managed objects. In contrast, an entity that is only a value has no additional semantics – it is used to simply supply data (e.g., a measured or observable value) for one or more managed entities to which it applies.

In DEN-ng, a *ManagedEntity* is *something of interest that can be managed*. Typical examples include *Products* (e.g., an application suite), *Resources* (e.g., network devices, and computers), and *Services* (e.g., VPNs and protocols). Any *ManagedEntity* can have an associated context that allows it to take on functionality specific to that context (using, for example, roles). DEN-ng currently is being used to refactor data from MIBs as well as represent CLI commands. However, note that MIB and CLI do not represent network “semantics” (e.g., only use this type of queuing if a CPU running at a certain minimum speed with at least X MB of memory), nor have they tried to represent non-network data, such as customer data or business rules. UML is inherently limited in what it can be used to represent, since it was designed for software engineering, not knowledge engineering. For example, it has no ability to represent common linguistic relationships (e.g., a synonym) or to reason, because it has no logic capabilities. Hence, it cannot relate the semantics of different languages and vocabularies to each other (e.g., to relate the commands of different network devices to each other, in order to realize a common end-to-end service).

4 Strengths and Weaknesses of Using Ontologies

In computer science and knowledge engineering, ontologies define theories of what exists. [14] defines an ontology as follows:

An ontology is a formal, explicit specification of a shared, machine-readable vocabulary and meanings, in the form of various entities and relationships between them, to describe knowledge about the contents of one or more related subject domains throughout the life cycle of its existence... Formal refers to the fact that the ontology should be representable in a formal grammar. Explicit means that the entities and relationships used, and the constraints on their use, are precisely and unambiguously defined in a declarative language suitable for knowledge representation. Shared means that all users of an ontology will represent a concept using the same or equivalent set of entities and relationships. Subject domain refers to the content of the universe of discourse being represented by the ontology.

An ontology language is made up of three components – syntax, semantics, and proof theory. The syntax of an ontology language is itself divided into three areas: the logic lexicon, the non-logic lexicon, and the grammar. The logic lexicon defines the set of connectives and quantifiers that can be used. Typical examples include the five basic connectives “and”, “or”, “not”, “if”, and “iff” (if and only if) and the two basic quantifiers: universal (“for all”) and existential (“exists”). The non-logic lexicon defines the vocabulary and any other associated elements of the vocabulary. For example, it may be convenient to separate the vocabulary and auxiliary knowledge that augments the vocabulary, such as definitions, “see” and “see also” meanings (as used in thesauri), and relationships. Finally, the grammar rules define how sentences are constructed.

Semantics (also called model theory) aims at providing a minimal formal description that enables the truth or falsity of any expression to be established. This includes mapping from expressions in the syntax to elements in the interpretation, assigning a truth value to atomic expressions in terms of the semantic values of its constituent names and predicates, and rules to define the truth of expressions in terms of the truth of its sub-expressions.

The proof theory consists of a set of syntactic methods that can be used to infer new sentences. Examples are modus ponens and various resolution and elimination algorithms.

The Semantic Web [15] has been described as “a new form of web content that is meaningful to computers will unleash a revolution of new possibilities.” Most of the Web’s content today is designed for humans to read, not for computer programs to manipulate (let alone understand and reason about!). This is because computers cannot process the semantics that are either implicitly associated with web content, or are not associated but could be inferred. The Semantic Web is not a separate Web; rather, it is an extension of the current one, in which information is given well-defined meaning, better enabling people to explicitly and implicitly use common semantics that computers can understand.

Ontologies are one popular way to define meaning, and are starting to be used heavily in the Semantic Web. Two popular ways of constructing an ontology are (1) using a language based on RDF [16] or RDFS [17], such as OWL [18], and (2) using a Description Logic [19]. However, there is currently no widely accepted knowledge representation that a majority of people use. Without such a common knowledge representation, it will be impossible to fuse information from documents and data sources written in different languages that use different knowledge representations.

Despite the efforts of the W3C, the use of RDF and RDFS has some unusual syntactic and semantic aspects, and hence may impair this effort. For example, [20] points out that “Semantic Web Architecture in which languages of increasing power are layered one on top of the other. Unfortunately, the relationships between adjacent layers are not specified, either with respect to syntax or semantics”, and concludes that RDF/RDFS is *not* the “best solution” for layering.

There is a profound need for describing resources and services, as well as *knowledge* about their use and the environment in which they are operating, in an unambiguous, computer-interpretable, extensible manner. Furthermore, device vendors will never openly embrace *retooling* their existing languages and management data, as there is no compelling *business reason* to do so. Hence, instead of trying to define and force acceptance of a new network management language, our approach defines a new knowledge representation that vendor- and device-specific management data can be *mapped to*.

5 Our Novel Knowledge Engineering Approach

Our knowledge engineering approach is based on representing facts (from data and information models) that are augmented with additional semantics (from ontologies). To do this, we construct a graph, whose nodes consist of model elements and ontology concepts that are joined by edges that represent their semantic similarity. This is part of a larger effort to develop a new knowledge representation and associated language, which will be described in a future paper.

Figure 2 shows an overview of our approach. In step 1, the information (or data) model(s) and the ontology (or ontologies) are each represented as graphs, which enable us to define semantic relationships to link nodes from one graph to nodes of the other graph. Since the data in the models and ontologies will usually be of varying granularity and use different terminology, Step 2 maps candidate model elements (N_M) to a Universal Lexicon, which serves as a common mapping vehicle that defines a shared, consensual set of all words, phrases and symbols used in the system. Each word, phrase, or symbol can, in general, have a set of meanings; hence, the best or most appropriate meaning of each word, phrase or symbol can be chosen given the correct context. Candidate model elements are those that have associated metadata (i.e., are not just “values”), since these model elements are *expected* to have associated meaning. Each successful mapping from a model element to a term or set of terms in the Universal Lexicon defines a possible candidate term that can be further described by one or more ontologies. Step 3 uses this information to select a subset of ontology nodes, N_O , from the set of ontologies for this particular domain. This forms a set of edges, $E_{M \rightarrow O}$, that represent concepts that are related to model elements. Step 4 augments this by defining a set of linguistic relationships that relate each unique concept in the set of ontology nodes N_O to one or more model elements N_{IM} . This is realized as a set of edges $E_{O \rightarrow M}$, which represent model elements related to concepts. Each of these edges can be thought of as a pointer that *augments* the fact modelled in the information model with additional knowledge that exists in the set of ontology nodes that were identified by those edges or vice-versa.

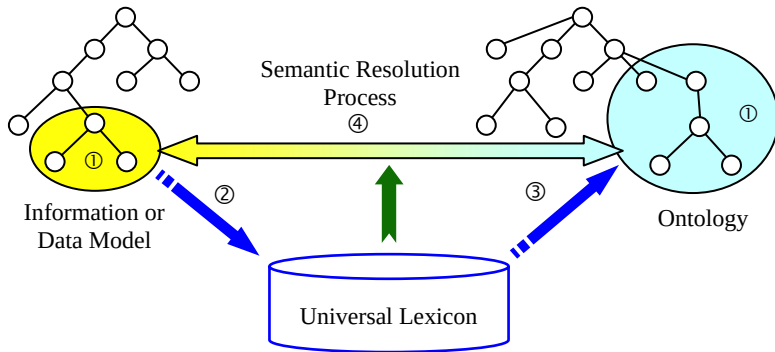


Fig. 2. Overview of Our Knowledge Approach

During the above four steps, additional relationships and concepts will be discovered. For example, a set of edges $E_{M \rightarrow O}$ can be defined that identify model elements with concepts in one or more ontologies. Any one of the concepts that are identified by these edges can be semantically related to additional concepts that may or may not be directly related to the original set of model elements. For example, referring to Fig. 3, the original sub-graph in the information model (three nodes and two edges, shown in an ellipse) is mapped to a sub-graph in the ontology consisting of five nodes and three edges.

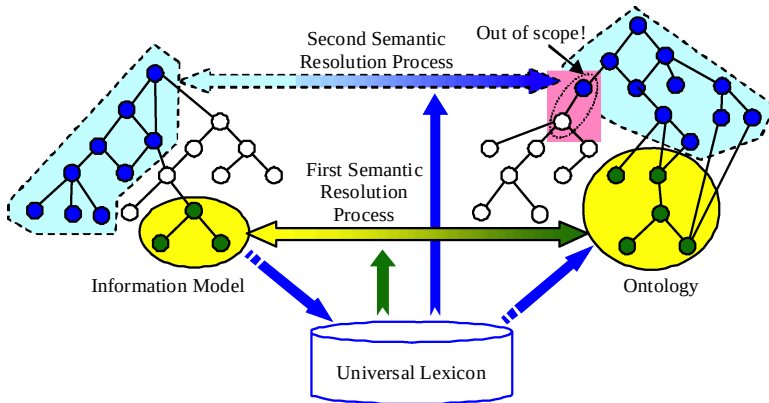


Fig. 3. Recursive Nature of the Semantic Mapping Process

However, some of the identified concepts in the ontology are in turn related to other concepts. Hence, we follow these relationships and construct a more complete graph, shown in the dashed polygon in Fig. 3. Given these new concepts, the information model is again examined to see if there are any additional model elements that match these new ontology concepts. This results in a new sub-graph (again shown in a dashed polygon). This recursion continues until no new elements from either the ontologies or the models are identified. The end data structure will contain

information from *all* of these nodes and edges. One additional relationship in the ontology, shown in a dotted ellipse, was also identified but excluded from this mapping. This is because while the concept in this dotted ellipse was related to the concept in the dashed polygon (which is part of the mapping!), the *semantic strength* of this relationship fell below threshold. This is beyond the scope of this paper; for a discussion on similarity algorithms, please refer to, for example, [21].

6 Initial Experiments

Experiments were conducted using the beta version of Protégé [22] in combination with the RACER PRO reasoner [23]. The following experiment is indicative of the power of this approach.

One of the most important problems in network management is to relate network problems (e.g., as indicated via SNMP alarms) to SLAs of customers. There is no way to directly accomplish this in SNMP or CLI, since neither contains any notion of a customer or an SLA. Our approach solves this problem by inferring which alarms are related to which SLAs of which customer through semantic reasoning.

In our approach, we use the information model as both a design guide and as an object instance repository. This latter and novel use is key to our model-driven management, as it enables changes in the model to be directly reflected in the code that is generated from the model. The model enables SNMP alarm instances to be directly related to the equipment instances that generated the alarms. At this point, we can use a number of different semantic relationships to establish a link between that equipment and a customer. For example, a particular IP address may be able to be mapped to a given geographic location, and that location identified as the customer's location. Or, that IP address could be part of a particular Virtual LAN that serves a particular customer. Or, a report of a customer problem can be associated with a network problem, which is a superclass of an SNMP alarm. In any of these cases (and others), once the customer is identified, it is a simple manner to match the traffic that the alarm referred to with one or more SLA parameters.

Fixing the problem is also aided by our approach. Suppose that the customer connection is supported by two different devices that each use different programming languages (e.g., vendor-specific SNMP and CLI). In this case, the commands of each device can be represented by graphs, which enable us to find the set of commands that have the closest semantic similarity to another set of commands.

Our solution is efficient, as all relationships can be pre-computed before the system is running. This is important, because the loading and pre-computing of relationships for large numbers of individuals becomes significant. For example, in a simple network management scenario, the memory required by RacerPRO was almost 570MB; however, its response is quick, as it runs in memory. Query performance varies as a function of the complexity of the query; in particular, if ontology role assertions are required, then RacerPRO (or *any* reasoned based on description logic) becomes unfeasible, as calculation times grow exponentially. Query response ranges from 0.3 seconds to 24.5, depending on the complexity of the query and the depth and breadth of the search required. For large networks, this will grow to be hours, but that is comparable or better than current tools, which are always used in an offline mode.

Note that an additional benefit of our approach is that it can be used to dynamically recognise *new* or *changed* knowledge. New knowledge is identified by noting that the input data does not match any model element (and possibly does not match any concept, though since ontologies are by nature more generic, this is less likely). Changed knowledge results in either not instantiating an expected association in the model, or in instantiating different elements from the model or ontologies.

7 Summary and Future Work

This paper has outlined a novel knowledge engineering approach that combines knowledge from information and data models with knowledge from ontologies into a graph. Conceptually, facts (which are represented by model elements) are augmented with additional meaning (which is represented by ontological concepts). This approach is amenable for computation before the system being managed is run, so that the only overhead incurred is matching sensor data to pre-defined nodes and edges in the graph. This has been verified through several different implementations.

Prior art searches have revealed no similar approaches. This is most likely because there are very few developers that are skilled in the areas covered by our approach (namely, logic programming, network management, and business process modelling). This is not surprising, as most network management engineers work at a significantly lower level of abstraction, whilst most business modellers work at a very different level of abstraction.

Autonomic networks require dynamic knowledge bases that are updated to reflect new and changed knowledge. Future work includes refining our approach to cost-efficiently recognise new and/or changed knowledge. Future work will also address associating new concepts and terminology from existing ontologies and other knowledge sources with our lexicon, so that our system can be more easily repurposed to serve the needs of other applications. This will be further developed into a new knowledge representation and language that can be used to automate semantic reasoning. In addition, we will also work on examining both the structure of the ontology used as well as the internal structure of the tableau algorithms used by RacerPRO in order to optimise its performance.

Acknowledgments. This work has received support from Science Foundation Ireland under the “Autonomic Management of Communications Networks and Services” award (grant no. 04/IN3/I404C).

References

1. Harrington, D., Preshun, R., Wijnen, B.: An Architecture for Describing Simple Network Management Protocol Management Frameworks, RFC3411, STD0062
2. For an example of a widely deployed CLI, Cisco, <http://www.cisco.com/warp/cpropub/45/tutorial.htm>
3. IBM, An Architectural Blueprint for Autonomic Computing, v7 (June 2005), <http://www.ibm.com/developerworks/autonomic/library/ac-summary/ac-blue.html>

4. Kephart, J., Chess, D.: The Vision of Autonomic Computing (January 2003), http://www.research.ibm.com/autonomic/research/papers/AC_Vision_Computer_Jan_2003.pdf
5. Strassner, J.: Autonomic Networking – Theory and Practice. In: 20th Network Operations and Management Symposium (NOMS) 2008 Tutorial, Salvador Bahia, Brazil (2008)
6. Strassner, J., Agoulmine, N., Lehtihet, E.: FOCALÉ – A Novel Autonomic Networking Architecture. *International Transactions on Systems, Science, and Applications (ITSSA) Journal* 3(1), 64–79 (2007)
7. Strassner, J., Samudrala, S., Cox, G., Liu, Y., Jiang, M., Zhang, J., van der Meer, S., Foghlú, M.Ó., Donnelly, W.: The Design of a New Context-Aware Policy Model for Autonomic Networking. In: 5th IEEE International Conference on Autonomic Computing (ICAC), Chicago, Illinois, June 2–6 (2008)
8. Cox, G., Serrat, J., Strassner, J., de Souza, J.N., Raymer, D., Samudrala, S., Jennings, B., Barrett, K.: An Enhanced Policy Model to Enable Autonomic Communications. In: 5th IEEE Workshop on Engineering of Autonomic and Autonomous Systems (EASE), pp. 184–193
9. Strassner, J.: Policy Based Network Management. Morgan Kaufman, San Francisco (2003)
10. <http://autonomic-communication-forum.org/>
11. Parsons, J., Wand, Y.: Emancipating Instances from the Tyranny of Classes in Information Modeling. *ACM Transactions on Database Systems* 25(2), 228–268 (2000)
12. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, Reading (1995)
13. Evans, E.: Domain-Driven Design: Tackling Complexity in the Heart of Software (2003); ISBN 0321125215
14. Strassner, J.: Knowledge Engineering Using Ontologies. In: Berstra, J., Burgess, M. (eds.) *Handbook of Network and System Administration*, vol. 4. Elsevier, Amsterdam (2008)
15. <http://www.semanticweb.org>
16. W3C, RDF Primer, W3C Recommendation (February 10, 2004), <http://www.w3.org/TR/rdf-primer/>
17. W3C, RDF Vocabulary Description Language 1.0: RDF Schema, W3C Recommendation (February 10, 2004), <http://www.w3.org/TR/rdf-schema/>
18. <http://www.w3.org/TR/owl-features>
19. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F.: The Description Logic Handbook: Theory, Implementation, Applications. Cambridge University Press, Cambridge (2003)
20. Horrocks, I., Patel-Schneider, P.F.: Three Theses of Representation in the Semantic Web. In: Proc. of the Twelfth International World Wide Web Conference (WWW 2003), pp. 39–47. ACM, New York (2003)
21. Wong, A., Ray, P., Parameswaran, N., Strassner, J.: Ontology Mapping for the Interoperability Problem in Network Management. *IEEE Journal on Selected Area in Communications* 23(10), 2058–2068 (2005)
22. <http://protege.stanford.edu>
23. Haarslev, V., Moller, R.: RACER system description. In: Goré, R.P., Leitsch, A., Nipkow, T. (eds.) *IJCAR 2001. LNCS (LNAI)*, vol. 2083, pp. 701–705. Springer, Heidelberg (2001)
24. SMI is defined here, <http://www.rfc-editor.org/rfc/rfc2578.txt>

Towards Management Requirements of Future Internet^{*}

Sung-Su Kim¹, Mi-Jung Choi², Hong-Taek Ju³, Masayoshi Ejiri¹,
and James Won-Ki Hong¹

¹Dept. of Computer Science and Engineering, POSTECH, Korea

²Dept. of Computer Science, Kangwon National University, Korea
{kiss,mjchoi,ejiri,jwkhong}@postech.ac.kr

³Dept. of Computer Engineering, Keimyung University, Korea
juht@kmu.ac.kr

Abstract. The Internet is one of the most successful modern technologies; we cannot imagine what our lives would be without the Internet. Despite the huge success of the Internet, many problems such as shortage of IP addresses, insufficient security and performance have emerged because of the increasing worldwide Internet population. Furthermore, the technologies developed to incrementally solve the abovementioned problems are causing new problems. Therefore, a redesign of the Internet architecture must be considered. Many research groups around the world are conducting work on the next version of the Internet architecture, called the Future Internet architecture. In this paper, we propose the management requirements and operations of the Future Internet with regard to the key parameters that are being researched through initiatives of research groups, such as FIND, GENI, FIRE and JGN.

Keywords: Future Internet, Management operation, Future Internet architecture, Cross-layer, resource virtualization, overlay network.

1 Introduction

The Internet is a social phenomenon that has changed and continues to change how humans communicate, work, conduct businesses, handle emergencies, process military operations, etc. It has redefined expectations on the possibilities of interactions between humans, between computers and humans, and between computers. Almost all the major industrial sectors have readily accepted and utilized the Internet for its advantages. This widespread use of the Internet reflects the extent to which it has been successfully implemented; however, it also hints at the complexity and diversity of a system that has grown from interconnecting merely a few supercomputers to interconnecting the world. Users of the Internet value the new and diverse set of applications on the basis of their interactivity, whereas developers of the applications value them according to the ease with which they can develop new functionalities and reach a large and diverse set of users [3].

^{*} This work was partly supported by the IT R&D program of MKE/IITA [2008-F-016-01, CASFI] and the EECE division at POSTECH under the BK21 program of MEST, Korea.

The success of the current Internet system is highlighted by the manner in which it has influenced our society. However, it is evident that the current Internet architecture is facing several challenges, many of them related to scalability issues, for example, supporting an ever growing number of users, devices, service attributes, applications, contexts, environments, security, vulnerability and networking technologies [2, 3, 4].

The worldwide research community has been increasingly drawn to the drawbacks of the current Internet. Many proposals have been suggested to solve the perceived problems, ranging from new enhanced protocols to fix specific problems up to the most radical proposal to redesign and deploy a fully new Internet. Most of the problems with the current Internet have originated from the tremendous pace of the increase in its use. Consequently, there was insufficient time available to address the drawbacks of the current Internet architecture. Many networking researchers around the world have identified the emerging limitations of the current Internet architecture and they have agreed that research must be conducted into the long-term applications of the Internet, and its basic architecture must be reconsidered. Furthermore, investigations must be performed to determine a better architecture, even if it does not appear backwards compatible.

New technological solutions may follow either an incremental approach or a “clean slate” approach to improving the Internet [1, 9]. The first approach involves the evolution of a system from one state to another by implementing incremental patches; the second approach demands a radical redesign of the Internet architecture to offer new abstractions and improved performance. The latter approach might prove to be disruptive with regard to technologies, services, and business models. We name this redesigned Internet architecture as the Future Internet architecture. In the past 30 years, the Internet architecture was successfully developed by using the incremental approach [9]. At present, we have reached a stage where people are unwilling or unable to experiment on the current architecture and want to apply the clean-slate approach.

To design the new Internet architecture, we must consider the manageability of Future Internet starting from scratch [1, 5, 7]. In this paper, we investigate the worldwide research conducted on the challenges, requirements, architectures, and management of Future Internet. On the basis of our investigations, we propose an integrated architecture for Future Internet and summarize the management requirements and operations of Future Internet. In Section 2, we summarize the research conducted on Future Internet in the U.S., EU, and Japan. In Section 3, we investigate the research studies conducted on the architectural aspects of Future Internet and we propose an integrated architecture for Future Internet. In Section 4, we illustrate the management requirements and operations of Future Internet. Finally, in Section 5, we provide the conclusion and discuss possible future studies.

2 Summary of Research on Future Internet

Research on Future Internet is performed worldwide, particularly in the U.S. [2, 13, 14], EU [3, 4, 15, 16], and Japan [17, 18]. Two types of research activities are conducted: design of a new network architecture and construction of an experimental facility to verify the proposed architecture.

In the U.S., the National Science Foundation (NSF) [12] is actively conducting research studies on Future Internet with regard to two perspectives; design and test facility. Two representative programs—Future Internet Design (FIND) [13] and Global Environment for Networking Innovations (GENI) [14]—are being conducted. FIND is a major long-term initiative of the NSF NeTS research program, it was started in 2006. It is also a funded project aimed at designing a next-generation Internet called the “Future Internet.” The research goal of FIND is to design an end-to-end network architecture as well as consider the implications of the emerging technologies on Future Internet.

GENI is an experimental facility/infrastructure set up to validate and demonstrate research studies, this facility was launched in August 2005. GENI consists of two components—GENI research program and GENI research facility. The GENI research facility is a state-of-the-art, global experimental facility that fosters the exploration and evaluation of new networking architectures (at scale) under realistic conditions.

Research on Future Internet was started in the European Union (EU) as the sixth Framework Program (FP6). In 2007, the seventh Framework Program (FP7) was started as a continuation of FP6. The current FP, FP7, will continue for seven years (2007–2013). Approximately € 600M has been earmarked for R&D in the field of networked systems, including technologies for Future Internet and Experimental Facilities [4].

Future Internet Research and Experimentation (FIRE) [15] is an initiative that is aimed at determining and consolidating research activities at design of Future Internet and various network testbeds in the European Union. The activities of FIRE are being launched in the 2nd ICT Call for proposals and projects are envisaged to start by early 2008, under the FP7-ICT Objective 1.6 “New Paradigms and Experimental Facilities” (budget: € 40M).

In Japan, research of Future Internet is divided into mainly two phases. The first phase is the Next Generation Network (NXGN) and the second phase is the New Generation Network (NWGN) [18]. In NXGN, the fundamental structure of IP networking is maintained and the Quadruple-play services must be provided. In NWGN, the network architectures and service conditions are different from those in

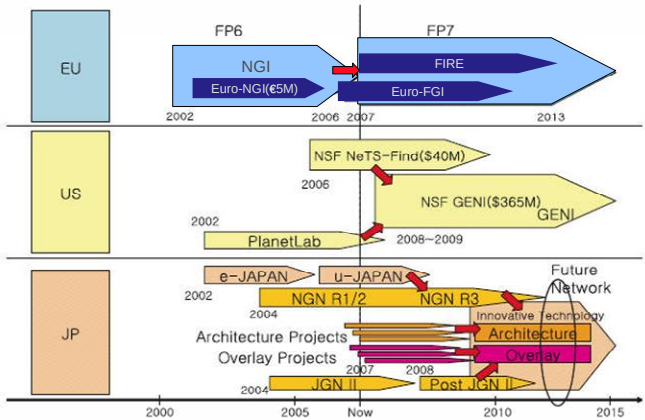


Fig. 1. Roadmaps of the Future Internet in the EU, U.S., and Japan

IP networks; hence, it supports a new network paradigm. At present, NWGN is in the research phase. Network architectures should be studied on the basis of the requirements for ubiquitous networking and new networking technologies such as advanced optical networking technologies.

Fig. 1 shows the summary of the research conducted on the Future Internet system in the EU, U.S., and Japan. The EU will continue through FP7 up to 2013. The FIRE initiative by the EU is involved with the design of a network architecture and construction of a test facility. The US started the construction of a test facility called PlanetLab in 2002, and continued the design of the Future Internet system through FIND; this will facilitate the upgradation of PlanetLab to GENI in order to build test environments on the basis of the design principles of FIND. Japan started the e-JAPAN and u-JAPAN projects in 2002 and 2006, respectively. Further, Japan started the projects to design an Internet architecture called New Generation Network (NWGN). Projects on overlay networks called JGN2 were started in 2004, and these projects will be continued by the name Post-JGN2 up to 2012.

3 Architecture of Future Internet

We have surveyed a large amount of research on the architecture of Future Internet, and we have extracted some key concepts such as cross-layer [10], resource virtualization [11], service-oriented architecture (SOA), and overlay network [17]. With regard to the challenges and requirements of Future Internet, we describe its architecture in this section.

Figure 2 shows our proposed architecture of Future Internet; it includes the key parameters of the Future Internet architecture, such as resource virtualization, cross-layered architecture, and service-oriented architecture, as investigated by other researchers [10, 11, 17].

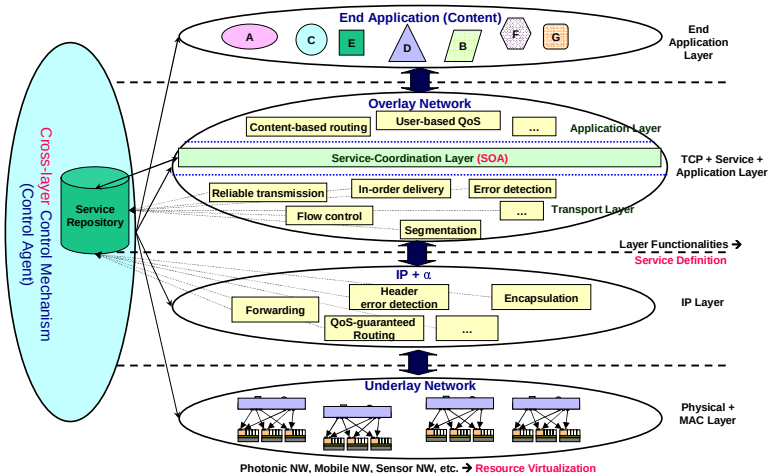


Fig. 2. Proposed Architecture of Future Internet

As shown in Figure 2, the proposed architecture is composed of four layers—underlay network, IP layer, overlay network, and applications. The key difference is in the overlay network. In the Japanese NWGN architecture [17], the overlay network will be focused on the extra functionalities, beyond what is supported by the basic internet. Similar to current internet, Future Internet will also need more functionalities. Overlay can provide, included mobility, customizing routing, QoS, novel addressing, enhanced security, multicast, and content distribution.

The functionalities of each layer are implemented as services, which are registered in the service repository. If a specific service is necessary, the service consumer requests the service repository by using a “find” operation, and the service repository sends the service provider information to the service consumer, then, the service consumer directly connects to the service provider and they interact with each other. This is the mechanism of the service-oriented architecture (SOA); the proposed architecture follows the mechanism of the SOA. To implement the SOA among various service providers and consumers, a new scheme of naming and discovery must be provided. Moreover, the scalability problem of registering, searching, and receiving services must be solved.

The current layered architecture must be transformed into a cross-layered architecture [10]. If the overlay network does not require the IP layer services, it directly calls the underlay network services through a cross-layer control agent. The cross-layer control agent plays a role in connecting the cross-layer communications over the adjacent layers. To support cross-layer communications, the cross-layer agent must support an integrated cross-layer interface.

The underlay network can be of various types, for example, wired optical networks and wireless sensor networks. First, we must identify the network resources and virtualize the network resources in order to support various user-oriented services and share network resources among the many customers and services.

4 Management of Future Internet

In this section, we investigate the research conducted on the management of Future Internet and we propose the management requirements and operations of Future Internet.

4.1 Management Requirements of Future Internet

On the basis of the research conducted on the management of Future Internet, we present the requirements for managing it in order to operate and administer network resources and services [7]. We divide the management requirements in accordance with the following management models: information model, communication model, functional model and non-functional model.

The management requirements of the information model are as follows:

- *High-level, goal-directed specification of the network properties and policies:* Human network managers should describe the required configuration properties. Software network managers should be responsible for translating high-level properties (goals) into strategies for implementation at lower levels [19].

- *Definition of various management objects*: The management information model must specify the management objects ranging from hardware (HW) resources to business goals and management functionalities.
- *Extensibility*: The management information model must be extensible so that it can be applied to any type of new network resource and service.

The management requirements of the communication model are as follows:

- *Support for management operations*: The communication model must provide the basic management operations such as get, set, create, add, delete, act, and notify.
- *Operationally independent, self bootstrapping management plane*: The management plane should be operationally independent of the data plane and should be able to bootstrap without any pre-configuration [19].
- *A single, simple management interface for all data plane protocols*: The operational complexity of protocols should be confined to their implementation, and they should express the information required for managing them through a simple management interface. This includes the responsibility on the protocol implementer for a detailed understanding of the protocol operation while reducing the burden on management applications.

The management requirements of the functional model are as follows:

- *FCAPS*: The functional model must support the basic management functionalities such as Fault, Configuration, Accounting, Performance and Security (FCAPS).
- *Support service-oriented architecture (SOA)*: As mentioned in Section 3, the architecture of Future Internet is based on the SOA. The management functionalities must be defined as services, registered in the service repository, and then discovered and implemented by using the SOA.
- *Intelligent and programmable network nodes*: Network nodes such as terminal, intermediate and core networks must be programmable in order to provide intelligent and autonomic management capabilities.
- *Identification and addressing*: The identities of users and objects (terminals, sensors, actuators, all types of electronic device), application identifiers, flows identifiers and network addresses (locations) may have to be defined independently by means of a global unified architectural approach. Such an address de-layering approach would support migration from multi-service networks to multi-network services, with services implemented by external facilities.
- *Discovery and Dissemination*: Each NMS periodically floods the network with beacons, which gather the IDs of the devices through which they propagate, thus they provide each device with a path to the NMS. This allows the devices to send source-routed frames to the NMS. In turn, the NMS can respond to these devices; this forms the NMSs *management channel*. Note that the management channel does not require any pre-configuration and is completely independent of the data plane paths in the network.
- *Data repository management*: Management of the Future Internet architecture will require data on the current state of the network, available in real time. The

challenge is that the proposed instrumentation systems can potentially gather vast quantities of high-dimensional data. This implies the requirement of a repository unit that will organize the measurement data efficiently and enable management decisions over a wide range of time scales.

- *Guaranteed QoS*: Currently, most ISPs employ over-provisioning methods to guarantee QoS. Although the Future Internet architecture is supposed to provide higher bandwidth and more cost-effective channels as compared to its predecessor networks, the cost of bandwidth in Future Internet wireless networks will remain higher than that in wired networks. Thus, over-provisioning methods in Future Internet will not be feasible and QoS support mechanisms will definitely be required [6].
- *Support generalized mobility*: Currently, horizontal handoff, which involves a terminal device to change cells within the same type of network in order to maintain service continuity, is the only service provided. In the future, mobility will be offered in a broader sense where users may have the ability to use technologies that are more accessible, thereby allowing movement between public wired access points and public wireless access points of various technologies. In Future Internet environments, in addition to horizontal handoff, vertical handoff must be supported. The greater challenge lies in facilitating vertical handoffs while satisfying a certain set of QoS requirements.

The non-functional management requirements of Future Internet are as follows:

- *Scalability*: The largest networks comprise thousands of routers/switches and tens of thousands of devices, and the default-free zone of the current Internet handles and routes hundreds of thousands of destination prefixes. Is it possible for conventional servers to manage a large number of devices and respond to events with sufficient speed in order to satisfy the network's goals? Will the amount of management information being transferred by the dissemination plane overwhelm the network's ability to carry data? Scalability is one of requirements of Future Internet. In addition, the management systems and operations of Future Internet must be scalable in order to support thousands and millions of different network devices and to provide services [5].
- *Interoperability*: We assume that there may exist different NMSs that play different roles such as security and performance. Also, interactions occur between NMSs in different domains, and the NMSs of different vendors compete for the same role in a given domain. Thus, interoperability needs to be provided.
- *Efficiency*: Along with the expansion of networking technologies and capabilities, there is an ever increasing need for streamlining and economizing network operations. Network management teams in Future Internet must be able to operate and maintain their infrastructures with the same amount or fewer human resources as compared to that required for the current networks [20].
- *Autonomic aspect*: Autonomic management has recently emerged as the evolution of automated management. In terms of management functionality, systems can be characterized as unmanaged, managed, predictive, adaptive, and autonomic [7]. Adaptive closed-loop automated management is the first

form of autonomic behavior, and the ultimate target is a fully distributed adaptive automated management. The various and complex network resources and services of Future Internet must be managed automatically and must include self-management capabilities.

4.2 Management Operations of Future Internet

Management of Future Internet involves providing management functions for the Future Internet resources and services and facilitating communication between the management planes and the network resources or services and other management planes. The FCAPS management functionalities must also be supported for the management of Future Internet [21]. These categories are sufficient to cover most, if not all, of the issues related to the operations and management of the wired networks in Future Internet. With the introduction of wireless and cross-layered networks, a few additional categories, which could not be easily covered by FCAPS, had to be added. The additional categories include mobility management, virtual resource management, cross-layer management, and terminal management. FCAPS covers the basic functions in conventional network management and we think these functional management operations are also necessary in the management of Future Internet. Thus, we do not mention FCAPS again but focus on the additional management operations in this paper.

4.2.1 Mobility Management

Mobility management facilitates horizontal and vertical handoffs and roaming. Both horizontal and vertical handoff functionalities must be provided and a mobility management system is required to support them. A key challenge will be facilitating the handoff functionalities when changing a network for a session with certain QoS requirements. The QoS requirements must be satisfied in the new network in order to avoid problems. The question remains whether we should terminate the session or continue it with lower quality. Although the probability of selection of the latter option is high for most users, such decisions can be preset by the user in his or her personal profile.

4.2.2 Virtual Resource Management

To provide resource virtualization, we must explore the implications of virtualization of a central architectural component of the Future Internet architecture, with the explicit objective of encouraging diversity of end-to-end networks [18]. We must consider how virtualization can be integrated into the Future Internet architecture and how it can be delivered by using multiple cooperating organizations on a large scale and with sufficiently high performance in order to make it economically compelling. The virtualization management includes the development of a complete architecture for a diversified Internet system and an experimental demonstration of the major components of the developed architecture; further, it includes the development of the underlying protocols and mechanisms for enabling multiple end-to-end networks to co-exist within a shared infrastructure that is owned and operated by multiple organizations.

4.2.3 Cross-Layered Service Management

To provide cross-layered services, we need to support cross-layer interactions. The cross-layer control mechanism is responsible for appropriately adjusting all the service- and method-specific interfaces and facilitating cross-layer services [8, 10]. Composing a cross-layer service refers to determining the subset of services it contains, their order in the stack, and the method for implementing each service. The objective is to dynamically build a cross-layer service for each new connection. A cross-layer control system takes into account the QoS requirements of the application, current network resource availability, the precedence constraints among services, and any policy in effect at the time of application. Specifically, the control system should be possible to construct abstract representations of services in order to determine their properties and interactions.

4.2.4 Terminal Management

Terminal management corresponds to two subareas, namely terminal location management and terminal trace management. The terminal location information can be used to make QoS-related decisions by the operations and management system when a user wishes to use a particular service and when a certain set of QoS has been requested. Terminal trace management is important for detecting and tracing stolen terminals or frauds, which are currently one of the biggest challenges experienced by the service providers at present. Traced information can also play a key role in activities such as determining the root cause of a malfunctioning terminal, advanced trouble shooting, optimization of resource usage and quality, RF coverage control and capacity improvement, and dropped call analysis.

5 Concluding Remarks

The current Internet will continue to evolve as a support mechanism for many applications in our society, economy, and daily life [5]. The current Internet principles were not designed to address the future challenges posed by mobility, security, manageability, and scalability. It is the right time to start exploring new research ideas to enable the design of the Future Internet infrastructure. Many research activities are currently being performed to design the Future Internet and build test facilities. These R&D activities are largely opened to international cooperation and partnerships. Their first batch of proposals are being prepared for submissions; interested parties from all regions of the developed and developing world have an opportunity to join and participate towards the development of a common platform for networking the future global economy.

We believe there are huge opportunities for the research community to pursue a more revolutionary clean-slate approach to the problem of network control and management. If successful, this line of research could create an entire landscape of possibilities for networking researchers to deploy their ideas on real networks. Ultimately, data networks, equipped with new control and management protocols and software, could be simpler, more robust, more evolvable, and less prone to security breaches.

In this paper, we extracted management requirements and operations for manageability of the Future Internet. But we cannot directly apply them to real Future Internet infrastructures, because the details of suggested operations in this paper are general and conceptual in nature. For future work, we need to transform these conceptual operations into more concrete operations and evaluate these operations in order to manage the Future Internet.

References

- [1] Albert, G., et al.: A Clean Slate 4D Approach to Network Control and Management. *ACM SIGCOMM Computer Communication Review* 35(5) (October 2005)
- [2] Darleen, F.: US National Science Foundation and the Future Internet Design. *ACM SIGCOMM Computer Communication Review* 37(3), 85–87 (2007)
- [3] Anastasius, G., et al.: Future internet research and experimentation: the FIRE initiative. *ACM SIGCOMM Computer Communication Review* 37(3), 89–92 (2007)
- [4] Schwarz, J., et al.: Future Internet Research: The EU framework. *ACM SIGCOMM Computer Communication Review* 37(2), 85–88 (2007)
- [5] Thrasyvoulos, S., et al.: Future internet: fundamentals and measurement. *ACM SIGCOMM Computer Communication Review* 37(2), 101–106 (2007)
- [6] Hao, C., et al.: An integrated, distributed traffic control strategy for the future internet. In: *Proceedings of the 2006 SIGCOMM workshop on Internet network management*, Pisa, Italy, pp. 17–22 (2006)
- [7] Siekkinen, M., et al.: Beyond the Future Internet—Requirements of Autonomic Networking Architectures to Address Long Term Future Networking Challenges. In: *11-th IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS 2007)*, March 21–23, pp. 89–98 (2007)
- [8] Madan, R., Cui, S., Lall, S., Goldsmith, A.: Cross-layer design for lifetime maximization in interference-limited wireless sensor networks. In: *Proceedings of the 2004 IEEE INFOCOM Conference* (March 2005)
- [9] Anja, F.: Internet clean-slate design: what and why? *ACM SIGCOM Computer Communication Review* 37(3) (July 2007)
- [10] Dutta, R., Rouskas, R., Baldine, I., Bragg, A., Stevenson, D.: The SILO Architecture for Services Integration, control, and Optimization for the Future Internet, communications. The future networked society, Release 1 (December 2006)
- [11] Larry, P., John, W.: Overview of the GENI Architecture, GENI document, GDD-06-11 (January 2007)
- [12] National Science Foundation (NSF) (December 2007), <http://www.nsf.gov/>
- [13] Future Internet Network Design (FIND) (December 2007), <http://find.isi.edu>
- [14] Global Environment for Network Innovations (GENI) (December 2007), <http://www.geni.net>
- [15] Future Internet Research and Experimentation (FIRE) (December 2007), <http://cordis.europa.eu/ist/fet/comms-fire.htm>
- [16] The Future-Internet Initiative (December 2007), <http://www.future-internet.eu>
- [17] Japan Giga Network 2 (JGN2) (December 2007), <http://www.jgn.nict.go.jp/english/index.html>
- [18] Yoshihiko, K., et al.: JGN 2 (Japan Gigabit Network 2). In: *Proceedings of the International Symposium on Applications and the Internet Workshops (SAINTW 2006)* (2005)

- [19] Paul, F., Jay, L.: Towards Complexity-Oblivious Network Management, FIND project (December 2007), <http://www.nets-find.net/Funded/TowardsComplexity.php>
- [20] Paul, B., Suman, B., Cristian, E.: Design for Manageability in the Next Generation Internet, FIND project (December 2007), <http://www.nets-find.net/Funded/Manageability.php>
- [21] Choi, M.-J., Hong, J.W.-K.: Towards Management of Next Generation Networks. IEICE Transactions on Communications E90-B(11), 3004–3014 (2007)

Considerations on NETCONF-Based Data Modeling

Hui Xu and Debao Xiao

Institute of Computer Network and Communication, Huazhong Normal University
430079 Wuhan, P.R. China
xuhui_1004@hotmail.com
<http://inc.ccnu.edu.cn>

Abstract. With the emergence of NETCONF, it is expected that a separate effort to study NETCONF-based data modeling in the interest of next generation network management should be undertaken. In order to exactly measure the capabilities of management data modeling languages, this paper tries to improve the initial evaluation framework with a weight to each criterion. And it aims in utilizing the optimized framework to the case of NETCONF-based network management by checking the feasibility of XML Schema and YANG as a NETCONF-specific data modeling language. The result shows that, YANG may be better than XML Schema for the purpose of promoting the standardization progress of NETCONF-based network management.

1 Introduction

NETCONF-based network management is now under hot research, which has emerged as a promising approach to standardize XML-based network management for the sake of automation. NETCONF has overcome the weaknesses of SNMP, and it provides a better configuration of IP network devices due to the effective use of XML and related technologies. However, NETCONF-based data modeling is still under study, for the reason that the specification of NETCONF content depends on the implementations, considering the current proprietary nature of the configuration data being manipulated [1].

As for this case, the definition of Information Model (IM) and Data Model (DM) must be seriously considered in order to facilitate the standardization progress of NETCONF-based network management. The main difference between these two concepts is the level of abstraction [2]. IMs always model Managed Objects (MOs) at a conceptual level, while DMs are defined at a concrete level. That is to say, multiple DMs can be derived from a single IM [3]. In fact, both DMs and IMs are now being complemented by semantic models. Concretely speaking, the meaning of concepts used in network management domain and relationships existing between them are made explicit, and this meaning is defined in a machine-readable format, thus making it accessible to both software management components and humans [2].

For each network management model, a data modeling language becomes essential to describe the managed resources, so as to promote vendor-neutral interoperable communication between managers and agents. To the best of our knowledge, few studies have been done on describing the capabilities for data modeling languages in

network management domain. Reference [4] assembles a list of technologies relevant for network management, but the summary emphasizes a general understanding, not focusing on data modeling languages. Reference [5] [6] regard management data modeling languages as a term of lightweight ontology because they just define information of the management domain without definition of the axioms or constraints present in heavyweight ontology, which makes them difficult to reason, and then compare existing languages in terms of their semantic expressiveness, the evaluation on which is limited to only one facet. Reference [7] provides some of the lessons learned from the SMIng project, especially ones about protocol-independent data modeling, which need to be considered by designers of future data modeling languages for network management protocols.

Obviously, the work on evaluating data modeling languages becomes extraordinarily necessary, especially for network management based on the developing NETCONF protocol. The aim of this paper is then to apply a generic evaluation framework to measure the capability of two promising NETCONF-based data modeling languages named XML Schema and YANG in network management domain, and optimize the initial framework by adding a weight to each criterion in adapting to the immediate needs of NETCONF-based network management, with more focus on those protocol-specific ones.

The remainder of this paper is organized as follows. The initial evaluation framework is briefly introduced in Section 2, and XML Schema and YANG are presented in Section 3. Consequentially in Section 4, the proposed framework is applied to these two languages in order to measure their capability for NETCONF-based data modeling. In order to balance protocol-specific criteria and protocol-independent ones for the urgent requirements of NETCONF-based network management, initial framework is optimized and then used to evaluate XML Schema and YANG from a quantificational point of view in Section 5. Section 6 concludes this paper.

2 Initial Evaluation Framework

It becomes indispensable to put forward a generic evaluation framework for data modeling languages to level their capabilities in satisfying the requirements of next generation network management. Our proposed framework demonstrated in Reference [8] is based on a set of criteria, which are modeling approaches, interoperability, readability, data representation, conformance, extensibility and security considerations, all with brief explanations as follows.

2.1 Modeling Approaches

When considering data modeling languages, it is better, first of all, to take their corresponding modeling approaches for network management [9] into account. Four main data modeling approaches are data-oriented one, command-oriented one, object-oriented/object-based one and document-oriented one.

The data-oriented approach models all the management aspects through data objects, and at least two operations (“get” and “set”) need to be defined. The command-oriented approach defines a large number of management operations, specifying not

the details but the commands to get/set selected information, which is generally in accordance with the task-oriented viewpoint by operators. The object-oriented/object-based approach combines the former two approaches in view of integration. The document-oriented approach represents various management data of a device as a structured document.

2.2 Interoperability

With the development of next generation networks, management in an environment with a heterogeneous set of devices becomes a trend. Hence, the standardization of DMs for network management should work at a higher level, making them more close to IMs. Following this way, data modeling languages should accordingly provide interoperability, which is consistent with the understanding of MOs already learned by network operators.

- Protocol independence
- Naming independence

2.3 Readability

It is desired that management data modeling languages should be easily understood by both operators and computers. Human readability is necessary for convenient study and use, and machine readability is required to accelerate automatic process of network management.

- Human readability
- Machine readability

2.4 Data Representation

Traditional network management solutions are often not strong in configuration management mostly as a result of modeling problems related to data representation, such as configuration objects being not easily identified. Consequently, the level of data representation ability should be included.

- Diversity of data types
- Specification of configuration data, state data and statistics data

2.5 Conformance

When defining MOs, it is also quite necessary to describe machine-readable conformance for the DM.

- Backward compatibility
- Versioning
- Definition of event notification messages
- Definition of error messages

2.6 Extensibility

With increasing complexity of the Internet, there is a great need for data modeling languages to have the ability to extend data structures, data types, elements and attributes easily for the practice of developing related software or systems to manage future heterogeneous networks.

- Extensibility of data structures
- Extensibility of data types
- Extensibility of elements and attributes

2.7 Security Considerations

Security cannot be ignored by modeling languages to ensure confidentiality and integrity of management data.

- Granularity of access control
- Lock mechanism

3 NETCONF-Based Data Modeling Language Presentation

Although NETCONF happens to use XML encoding and optional XPath filtering, it is indeed a network management protocol. Consequently, the conceptual management data available on a device should be understood by operators. NETCONF needs its own data modeling language, just like SNMP needs SMIV2. Since NETCONF is designed to meet special requirements of the configuration, its data modeling language needs to support special features not found in other languages, especially those immediate requirement for NETCONF [10]. For example, it has to deal with different contexts such as *running*, *candidate* and *startup* configs, which respectively represent three different states of the configuration.

For one thing, two prospective NETCONF-based data modeling languages being measured, which are XML Schema and YANG, will be presented in chronological order with a brief introduction.

3.1 XML Schema

Due to the drawbacks of previous data modeling languages, IETF has taken XML-based data modeling into consideration, especially in IETF NETCONF Working Group (WG) [11]. Thus, NETCONF-based data modeling focuses on XML Schema and other XML-based languages.

Adopting a document-oriented approach, XML Schema [12] seems to be promising as a data modeling language for the purpose of standardizing NETCONF-based network management. S. Chisholm etc propose the use of XML Schema to define NETCONF content [13]. A framework for defining NETCONF content is then provided using a meta-model and XML Schema, which aims to build on existing well-deployed technologies and overlap management-specific semantics to ensure high-quality interoperable NETCONF content definition.

3.2 YANG

YANG [14] is a data modeling language based on the Structure of Management Information, next generation (SMIng) [15] [16] syntax, being used to model semantics and organization of configuration and state data manipulated by the NETCONF protocol, NETCONF Remote Procedure Calls (RPCs), and NETCONF notifications. In order to provide readability as the highest priority, YANG focuses on semantics and can be directed mapped to XML content.

Another advantage of YANG is that, it tries to fulfill all agreed requirements defined in Reference [10], and it is NETCONF-specific, aiming to support all the features of the NETCONF protocol, including all base operations, notifications and RPC operations. Although YANG is now limited in scope of usage, being applied only to NETCONF, it gains experience from existing implementations, maximizes utility within this scope and can possibly be extended in the future.

Currently, the new NETCONF Data Modeling Language (netmod) WG [17] proposed by IETF plans to use YANG as its starting point for the unified data modeling language to standardize the NETCONF content layer.

4 Evaluation Result

Using the evaluation framework demonstrated in Section 2, we compare XML Schema and YANG, the result of which is presented in Table 1. And our measurement except for modeling approaches is classified as the following four levels.

Level 1: A minus sign (–) means that the language does not have such a capability.

Level 2: An asterisk sign (*) denotes that the language is weak in this capability.

Level 3: A plus sign (+) is used when the language is good at this capability.

Level 4: Two plus sign (++) is placed when the language completely possesses this capability.

As is demonstrated in Table 1, it can first be summarized that, most properties of XML Schema surpass those of previous data modeling languages, especially in aspects such as interoperability, data representation and extensibility, which have been quite well-known by its numerous users. However, its machine readability is not so satisfying, for the reason that, what it is accomplished in is not semantic expressiveness but content definition. Furthermore, compared to its wide application, XML Schema is both too complicated and excessively general as a data modeling language for special use only in the scope of network management. On the other hand, definition of a NETCONF-based management DM is much more than an XML instance document description, or in other words, XML Schema is still not expressive enough with a view to NETCONF-based data modeling.

All these reasons above lead to the fact that, there are still no DMs defined by XML Schema yet. Currently, IETF Operations & Management (OPS) Area [18] is focusing the solutions on SMI-to-XML Schema conversion and XSD for accessing SMIPv2 DMs, since SNMP-based network management has been supplemented with a lot of proprietary MIB modules defined by different device vendors, and discarding MIB objects and SMI syntax when designing a new DM does reduce this benefit from experience of so many years. But more work has to be done on standardizing a NETCONF-based data modeling language for network management.

Table 1. Evaluation result of XML Schema and YANG through comparison

Criteria	XML Schema	YANG
Modeling Approaches		
Data-oriented		
Command-oriented		
Object-based		√
Object-oriented		
Document-oriented	√	√
Interoperability		
Protocol independence	++	—
Naming independence	++	—
Readability		
Human readability	+	++
Machine readability	*	+
Data Representation		
Diversity of data types	++	++
Specification of configuration data, state data and statistics data	++	++
Conformance		
Backward compatibility	+	+
Versioning	++	++
Definition of event notification messages	—	++
Definition of error messages	—	++
Extensibility		
Extensibility of data structures	++	++
Extensibility of data types	++	++
Extensibility of elements and attributes	++	++
Security Considerations		
Granularity of access control	++	++
Lock mechanism	—	++

Compared to XML Schema, it can then be seen from Table 1 that, YANG is a NETCONF-specific data modeling language, taking semantics such as notification message definitions, error message definitions and lock mechanism into consideration. All these features make YANG much easier to describe DMs in a way that maps to NETCONF in a very straight-forward manner, and therefore YANG has been chosen as the best approach up to now [14], due to its focus on the immediate requirements for NETCONF-based network management.

Additionally, as a data modeling language for network management, YANG is designed at the semantic level, while XML Schema and RELAX NG are at the syntactic level, the relationship of which is illuminated in Fig. 1. That is to say, description by YANG is at a semantic level, and XSD can be generated from the DMs defined in YANG. Thus in this case, existing tools can be utilized.

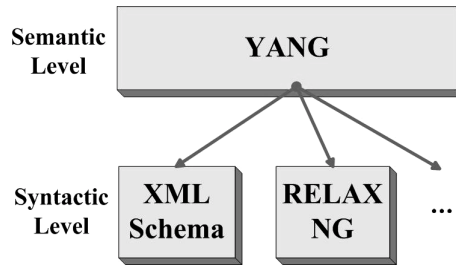


Fig. 1. YANG at the semantic level and XML-related languages at the syntactic level

Table 2. Optimized evaluation framework

Weight	Criteria	Domain
0.1	Modeling Approaches	0-1
	Data-Oriented	0-0.1
	Command-Oriented	0-0.1
	Object-Based	0-0.15
	Object-Oriented	0-0.3
	Document-Oriented	0-0.35
0.1	Interoperability	0-1
	Protocol Independence	0-0.8
	Naming Independence	0-0.2
0.25	Readability	0-1
	Human Readability	0-0.6
	Machine Readability	0-0.4
0.25	Data Representation	0-1
	Diversity of Data Types	0-0.4
	Specification of Configuration Data, State Data and Statistics Data	0-0.6
0.05	Conformance	0-1
	Backward Compatibility	0-0.3
	Versioning	0-0.1
	Definition of Event Notification Messages	0-0.3
	Definition of Error Messages	0-0.3
0.05	Extensibility	0-1
	Extensibility of Data Structures	0-0.3
	Extensibility of Data Types	0-0.3
	Extensibility of Elements and Attributes	0-0.4
0.2	Security Considerations	0-1
	Granularity of Access Control	0-0.4
	Lock Mechanism	0-0.6

Note that, YANG is weak in interoperability, as is indicated in Table 1, since it is now limited in applications only to NETCONF-based network management. However, YANG is desired to be extended in the future, which will promote its protocol independence.

5 Optimized Evaluation Framework

In order to make full use of the initial evaluation framework for the special purpose of NETCONF-based network management, a weight is selected and added to each criterion along with a domain to its facets, which guarantee that both the value of each criterion and the final value of the evaluated language are between 0 and 1. The optimized evaluation framework is demonstrated in Table 2.

Table 3. Application of the optimized framework to evaluate XML Schema and YANG

Criteria	XML Schema	YANG
Modeling Approaches	0.03	0.05
Data-Oriented	0	0
Command-Oriented	0	0
Object-Based	0	0.15
Object-Oriented	0	0
Document-Oriented	0.3	0.35
Interoperability	0.075	0.025
Protocol Independence	0.6	0.2
Naming Independence	0.15	0.05
Readability	0.075	0.2
Human Readability	0.25	0.5
Machine Readability	0.05	0.3
Data Representation	0.15	0.225
Diversity of Data Types	0.3	0.3
Specification of Configuration Data, State Data and Statistics Data	0.3	0.6
Conformance	0.025	0.045
Backward Compatibility	0.25	0.25
Versioning	0.1	0.1
Definition of Event	0	0.3
Notification Messages		
Definition of Error Messages	0.15	0.25
Extensibility	0.05	0.035
Extensibility of Data Structures	0.3	0
Extensibility of Data Types	0.3	0.3
Extensibility of Elements and Attributes	0.4	0.4
Security Considerations	0.04	0.08
Granularity of Access Control	0.2	0.4
Lock Mechanism	0	0
Sum	0.445	0.66

As is shown in Table 2, the value of weight to each criterion extraordinarily depends on the level of importance demanded by NETCONF-based data modeling, with a brief explanation as follows.

- Readability (especially human readability), data representation (especially specification of configuration data, state data and statistics data), and security considerations (especially lock mechanism) have been pointed out by NETCONF for the consideration of data modeling. Hence, their weights are marked high.
- Modeling approaches, interoperability, conformance and extensibility are criteria related to protocol independence for generic network management, thus are labeled as ones with low weights.

Table 3 presents application of the improved framework proposed above to measure the capability of XML Schema and YANG as a management data modeling language in the numeric form. As is indicated in Table 3, YANG is better than XML Schema in the interest of promoting the standardization progress of NETCONF-based network management.

6 Conclusions

This paper investigates issues on data modeling in the case of NETCONF-based network management through application of optimized evaluation framework based on the initial one universal for management data modeling languages. The presented approach not only provides a prospective solution for NETCONF-based data modeling, but also points out a new direction for future research on data modeling in next generation network management domain.

Acknowledgment

This work is partially supported by the Scientific and Technological Planning Project of Wuhan City, P. R. China under Grant No. 200710421130 and the Key Scientific and Technological Project of Hubei Province, P.R. China under Grant No. 2004AA103A01.

References

1. Enns, R. (ed.): NETCONF Configuration Protocol. RFC4741 (2006)
2. Pras, A., et al.: Key Research Challenges in Network Management. *IEEE Communications Magazine* 45(10), 104–110 (2007)
3. Pras, A., Schoenwaelder, J.: On the Differences between Information Models and Data Models. RFC3444 (2003)
4. Schoenwaelder, J., Pras, A., Martin-Flatin, J.-P.: On the Future of Internet Management Technologies. *IEEE Communications Magazine* 41(10), 90–97 (2003)
5. López de Vergara, J.E., Villagrà, V.A., Asensio, J.I., Berrocal, J.: Ontologies: Giving Semantics to Network Management Models. *IEEE Network* 17(3), 15–21 (2003)

6. López de Vergara, J.E., Villagrà, V.A., Berrocal, J.: Applying the Web Ontology Language to Management Information Definitions. *IEEE Communications Magazine* 42(7), 68–74 (2004)
7. Schoenwaelder, J.: Protocol-Independent Data Modeling: Lessons Learned from the SMIng Project. *IEEE Communications Magazine* 46(5), 148–153 (2008)
8. Xu, H., Xiao, D., Chang, Y., Chen, L.: An Evaluation Framework for Data Modeling Languages in Network Management Domain. In: *Proceeding of 3rd International MultiConference of Engineers and Computer Scientists*, pp. 1115–1120. LAENG Press (2008)
9. Pras, A.: *Network Management Architectures*. PhD Thesis, University of Twente (1995)
10. Presuhn, R. (ed.): *Requirements for a Configuration Data Modeling Language*. draft-presuhn-rcdml-03 (work in progress, 2008)
11. IETF: NETCONF Working Group, <http://www.ops.ietf.org/netconf/>
12. W3C: XML Schema, <http://www.w3.org/XML/Schema>
13. Chisholm, S., Clemm, A., Tjong, J.: Using XML Schema to Define NETCONF Content. draft-chisholm-netconf-model-08 (work in progress, 2008)
14. Bjorklund, M. (ed.): YANG - A Data Modeling Language for NETCONF. draft-ietf-netmod-yang-01 (work in progress, 2008)
15. Elliott, C.: SMIng Objectives. RFC3216 (2001)
16. Strauss, F., Schoenwaelder, J.: SMIng – Next Generation Structure of Management Information. RFC3780 (2004)
17. IETF: NETCONF Data Modeling Language Working Group, <http://www.ietf.org/html.charters/netmod-charter.html>
18. IETF: Operations & Management Area, <http://www.ops.ietf.org/>

A Hop by Hop Rate Control Based QoS Management for Real Time Traffic in Wireless Sensor Networks^{*}

Muhammad Mostafa Monowar, Md. Obaidur Rahman, Byung Goo Choi,
and Choong Seon Hong^{**}

Networking Lab, Department of Computer Engineering, Kyung Hee University
1 Seocheon, Giheung, Yongin, Gyeonggi, 449-701 South Korea
{monowar, rupam}@networking.khu.ac.kr, cshong@khu.ac.kr

Abstract. Wireless Sensor Network typically assimilates various real time applications that must meet some QoS requirements (e.g. delay, jitter, throughput, packet loss) under severe resource limitations. Hence, quality-of-service (QoS) management in Wireless Sensor Network is an important issue yet to be investigated. Due to the high data rate and burst traffic for real time applications, occurrence of congestion is very common. Ensuring the QoS requirements under congested scenario is quite challenging. As an attempt to this direction, in this paper we propose a hop-by-hop dynamic rate control algorithm which controls the congestion as well as ensures that the real time traffic will meet the soft QoS requirements. We have introduced per-hop deadline miss ratio as the congestion indication metric and performed the rate control when this ratio goes above to a certain level. Thus by per-hop rate adjustment, soft QoS is being met locally which in turns meets the end-to-end soft QoS. Finally, simulation has demonstrated the effectiveness of our approach.

Keywords: Congestion, Soft QoS, Per-hop deadline miss ratio, Real time traffic.

1 Introduction

The recent advances in wireless sensor network communication protocols [1] and low power hardware devices such as CMOS camera and microphones have elevated the proliferation of various sensor network applications i.e. battlefield surveillance, disaster and emergency response, environmental monitoring, industrial process control etc. These applications deal with a variety of real time constraints in response to the physical world. In comparison with the traditional distributed systems, real time guarantee for sensor network is more challenging due to the diverse transmission rates, unpredictable spatiotemporal properties of the physical events in the real world and the severe resource limitations in the sensor network.

In WSN, usually tens or thousands of sensor nodes are deployed in a scattered way in an area with single or multiple sinks. Due to the high data rate and unpredictable

^{*} This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD)" (KRF-2006-521-D00394).

^{**} Corresponding author.

burst traffic nature of the real time application, occurrence of congestion is more likely. Under congested scenario, the end-to-end meeting of deadline of the real time packets is seriously hindered. Hence, the required QoS cannot be ensured. Therefore, an efficient rate control mechanism is necessary that boost up the meeting of QoS requirements even in the congested scenario.

In this paper, we have proposed a QoS management mechanism which performs an efficient dynamic hop by hop rate adjustment to ensure the required QoS guarantee even in the congested scenario. We have used per-hop deadline miss ratio as the QoS metric as well as congestion detection metric. Lowering the per-hop deadline miss ratio, the meeting of end-to-end deadline has been ensured. We have also employed Per-hop Least Entailed Delay First (PLEDF) scheduler for reducing the deadline miss ratio. Here, we have considered the applications that require soft QoS which means it tolerates the delay and deadline misses up to a certain level.

The rest of the paper is organized as follows. Section 2 presents several related works on congestion control for real time traffic. Subsequently section 3 describes the design considerations for our proposed scheme. Section 4 represents our proposed protocol in detail. Section 5 describes the simulation and finally section 6 concludes the paper.

2 Related Works

In the present research train lots of works is going on the rate control for wireless sensor network. But a very few of them considered the QoS issues for the real time packets.

PCCP [2] is a recent congestion control protocol which takes into account the QoS of the multimedia applications. It introduces an efficient congestion detection technique addressing both node and link level congestion but can't guarantee the end-to-end deadline of the real time packets.

DART [3] is another current transport protocol for wireless sensor networks. The protocol simultaneously addresses the congestion control and timely event transport reliability. It is sink initiated protocol which controls congestion by adjusting the reporting rate that incurs extra delay for the sources far away from the sink.

SUPPORTS [4] is another real time traffic management protocols for Sensor Network. It is based on traffic regulation and end-to-end scheduling approach which uses hop-by-hop approach for traffic regulation and reject packets which is supposed to miss the deadline.

FLC-QM[5] is proposed as the Fuzzy logic based QoS Management in Wireless Sensor/Actuator (WSAN) Networks. It utilizes a fuzzy logic controller inside each source sensor node to adapt sampling period to the deadline miss ratio associated with data transmission from the sensor to actuator.

Besides these, CODA[6], CCF[7], FUSION[8] etc are the remarkable congestion control for wireless sensor networks but none of them explicitly considered the QoS for real time traffic.

In fact, the QoS issues considered for real time traffic in wireless sensor network mainly focuses on the routing layer solution such as SPEED [9], MMSPEED[10], Energy Aware QoS Routing [11] etc. RAP [12] is proposed to provide real time communication architecture for large scale wireless sensor networks. It utilizes MAC

prioritization to provide soft guarantees by locally considering the velocity at a node which can be dynamically readjusted. Hence, the need of an efficient protocol which provides QoS guarantees even in the congested situation motivates us to design our protocol.

3 Design Considerations

This section states the network model, assumptions and node model for our proposed protocol which we have taken into account while constructing our algorithm.

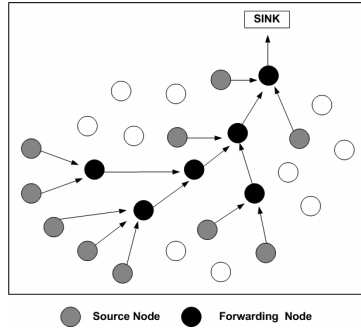


Fig. 1. Network Model

3.1 Network Model and Assumptions

In this paper, we consider the rate control for many-to-one multi hop single path routing. The network model is shown in Fig. 1 where the grey nodes function as a source node and black one as forwarding node. Source nodes transmit the real time packets periodically. All nodes are supposed to use CSMA/CA like MAC protocol. We assume that a predetermined route has been established by any routing protocol and path establishment is out of the scope of this paper. We further assume that the application will assign the deadline for every real time packet while originating. Nodes transmit its data to the downstream direction towards the sink are called child nodes and the node to which children transmits is their parent node.

3.2 Node Model

Fig. 2 depicts the node model for a particular node i . We have provisioned a QoS Management Module between the Network and MAC layer which ensures the lower deadline miss ratio with efficient rate adjustment even in the congested situation.

This module contains a Queue, Per-hop Least Entailed Delay First (PLEDF) scheduler, a Delay estimator and a Rate controller. The transit traffic comes from the previous node are placed into a queue after performing the scheduling by the PLEDF scheduler. Scheduling is done for each packet according to their least per-hop entailed delay with the assistance of delay estimator. In each node scheduler drops packets which are not able to meet their per hop deadline. Delay estimator is responsible to

provide the information to the scheduler about average waiting time within a node and per hop entailed delay for an incoming packet. The details of measuring the average waiting time and per-hop entailed delay have been presented in the section 4.1. The rate controller carries out the rate adjustment function as described in section 4.2 interacting with the MAC layer and scheduler.

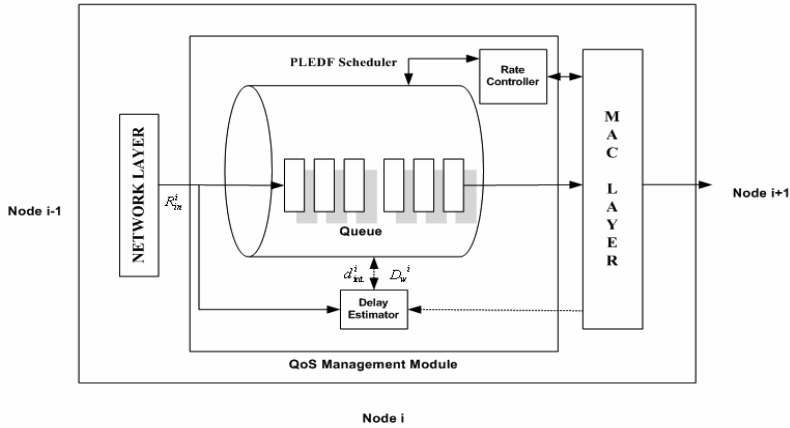


Fig. 2. Node Model

4 Proposed Protocol

Our main motivation in designing the protocol is to perform rate control for real time traffic transited through a single sensor node in such a way that a required number of packets must meet their end-to-end deadline. To accomplish this goal, every node maintains the deadline miss ratio for each of the real time packets passes through it in such a way that the deadline miss ratio will not exceed a certain level even in the congested situation. The details of the protocol are presented in the following subsections:

4.1 Determination of Per-Hop Deadline Miss Ratio

We have introduced per-hop deadline miss ratio as the basis of the rate adjustment. This is the most important metric for the soft real time system. This metric reflects the level of contention and congestion for a particular node. To verify this we have performed a simulation in ns-2 over 100 nodes using AODV as a routing protocol and we set the value of end-to-end deadline as 300 ms.. End-to-end deadline miss ratio is measured at the sink under different congested scenarios. Fig. 3 illustrates our result. Here, it shows that as the congestion increases (increasing the packet transmission rate) the deadline miss ratio also increases.

This is inevitable that the increase in congestion incurs the packet waiting delay at the buffer. Moreover, as the contention increases the packet transmission delay also increases. Hence, we have used this metric for the detection of congestion for the real time traffic on a particular node.

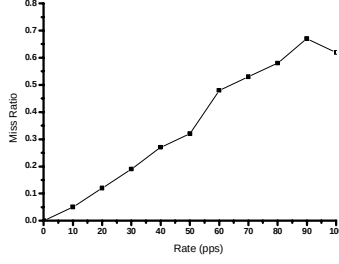


Fig. 3. Miss Ratio in Different Congested Scenario

The per-hop deadline miss ratio, d_{miss}^i is computed periodically. At the end of an epoch, t , the d_{miss}^i is calculated as-

$$d_{miss}^i(t) = \frac{N_{miss}(t)}{N_{sch}(t)} \quad (1)$$

Here, $N_{miss}(t)$ is the number of packets missed the per-hop deadline at the end of epoch, t and $N_{sch}(t)$ is the total number of packets scheduled during this epoch.

Now, the question is how we can calculate the value of $N_{miss}(t)$. To determine $N_{miss}(t)$, two information are needed; i) The projected waiting time within a node i , D_w^i and ii) Per-hop entailed delay, d_{ent}^i .

Estimation of the waiting time for a packet within a node

The projected waiting time within a node i , D_w^i is defined as the sum of the expected queue waiting time, t_q^i , and the average packet service time, $\overline{t_s^i}$. The queue waiting time t_q^i is the time a packet experiences from the moment it enters into the queue until it reaches the head of the queue, and ready for transmission. We have modeled the queue at each sensor node as M/G/1 system. Hence, the expected queue waiting time, t_q^i can be estimated by using the well known Pollaczek-Khinchin mean value formula as follows:

$$t_q^i = \frac{R_{in}^i \times \overline{t_s^{i^2}}}{2(1 - R_{in}^i \times \overline{t_s^i})} \quad (2)$$

Where, R_{in}^i is the packet incoming rate at the queue, $\overline{t_s^{i^2}} = \sigma^2 + \overline{t_s^i}^2$ and c^2 is the distribution variance.

Packet service time is defined as the time when the packet is ready for transmission until the last bit leaves from a node. It includes channel busy time, DIFS, backoff,

RTS,CTS and packet transmission time. By using EWMA (Exponential Weighted Moving Average Formula), $\bar{t}_s^i$ is updated each time a packet is forwarded as follows:

$$\bar{t}_s^i = (1 - w_s) \times \bar{t}_s^i + w_s \times inst(t_s^i) \quad (3)$$

Where, $inst(t_s^i)$ is the instantaneous service time of the packet just transmitted and w_s is a constant in the range of $0 < w_s < 1$.

Thus, the estimated waiting time within a node i is,

$$D_w^i = t_q^i + \bar{t}_s^i \quad (4)$$

The calculation of the estimated waiting time within a node is performed each time when a packet arrives at the queue.

Estimation of the per-hop entailed delay for a packet

We have adopted the estimation of per-hop entailed delay for a packet as the procedure described in [4]. The per-hop entailed delay, d_{int}^i works as a threshold in order to determining the per-hop deadline miss ratio for a packet. It can be defined as the maximum allowable time a packet can stay in a node from the moment it arrives until the transmission ends. Hence, per-hop entailed delay, d_{int}^i can be computed as

$$d_{int}^i = t_{deadline} - t_{elapsed} - t_{e2e}^i \quad (5)$$

Where, $t_{deadline}$ is the deadline for a packet, $t_{elapsed}$ is the elapsed time for a packet (the elapsed time will be piggybacked in the packet by the upstream node) since it was initiated from the source and t_{e2e}^i is the projected end-to-end (node i to sink) delay which can be calculated as,

$$t_{e2e}^i = D_w^i + \sum_{j=i+1}^M t_{e2e}^j \quad (6)$$

Here, M stands for the number of hops up to the sink from node i . Each node maintains and feedbacks to the previous hop regarding expected delay, t_{e2e}^i from itself to the sink. And this feedback information can be propagated through the per frame acknowledgement packet used in 802.11.

When D_w^i and d_{int}^i are available to node i for an incoming packet, the packet will be treated as it will miss the per-hop deadline when the relation, $d_{int}^i < D_w^i$ becomes true. As the calculation of entailed delay includes the estimated end-to-end delay from a particular node i , it guarantees that the packet which misses per-hop deadline at node i will certainly going to miss the end-to-end deadline.

The scheduler will maintain a counter to determine the number of the per-hop deadline missed packets during epoch duration as well as counts the total scheduled packets in that epoch. These parameter values are fed to the rate controller in order to calculate d_{miss}^i according to equation 1.

4.2 Hop by Hop Dynamic Rate Adjustment

The hop by hop dynamic rate adjustment is performed by the child nodes of particular node i , through the overhearing of the parent's packet. Each node i will piggyback its current average packet service time, $\overline{t_s^i}$, total number of child nodes, C_p and deadline miss ratio d_{miss}^i value in its packet header. Because of the broadcast nature of wireless channel, all the child nodes of node i overhear the packet header parameters. The update of their transmission rate occurred based on the value of d_{miss}^i of parent node. The rate adjustment algorithm is shown in Fig. 4.

Algorithm:	Hop by Hop Rate Adjustment
Input:	Each node i
Output:	Packet Transmission Rate, R_{trans}^i
Initialization()	
	$R_{trans}^i = R_{trans}^{init}, d_{miss}^i = 0$
	Calculate_Rate ($\overline{t_s^{p^i}}, C_p, d_{miss}^{p^i}$)
	If $d_{miss}^{p^i} < 0.1$
	then $R_{trans}^{p^i} = \frac{1}{\overline{t_s^{p^i}}}, R_{trans}^i = \frac{R_{trans}^{p^i}}{C_p}$
	End if
	If $d_{miss}^{p^i} > 0.1$
	then $R_{trans}^i = \frac{R_{trans}^{p^i}}{C_p} \times (1 - d_{miss}^{p^i})$
	End if
	Return R_{trans}^i

Fig. 4. Rate Adjustment Algorithm

The algorithm works as follows:

Initially, every source node will set their PHY/MAC transmission rate to a small value R_{trans}^{init} and the d_{miss}^i is set to 0.

During the normal operation of the network, as long as the parent's per-hop deadline miss ratio, $d_{miss}^{p^i}$ remains less than 0.1, each child node i , updates their transmission rate as,

$$R_{trans}^i = \frac{R_{trans}^{p^i}}{C_p}$$

Where, parent's transmission rate, $R_{trans}^{p^i}$ will be measured as the inverse of its packet service time, $\overline{t_s^{p^i}}$. This is performed in order to maintain the fairness among all the child nodes.

Each of the child node i , will update their transmission rate when $d_{miss}^{p^i} > 0.1$, (per-hop deadline miss ratio becomes greater than 10%) as follows:

$$R_{trans}^i = \frac{R_{trans}^{p^i}}{C_p} \times (1 - d_{miss}^{p^i})$$

This indicates the rate decrease proportional to the per-hop deadline miss ratio. As updated value of deadline miss ratio above a certain level (10%) indicates congestion, the proportional decrease rate lowers the incoming rate to the parents. This certainly reduces the waiting time of a packet. Therefore, the per-hop deadline miss ratio will decrease which indicates the subsiding of congestion.

But truly, the congestion detection threshold depends on the QoS requirements of the corresponding application. Application with the requirement of very lower deadline miss ratio may lower the threshold.

5 Simulation

We have performed extensive simulations using ns-2 [13] to evaluate the performance of our protocol. The simulation parameters are described as follows: 100 sensors are randomly deployed in 100x100 m² sensor field. The transmission range of the sensors is 30 m. The maximum communication channel bit rate is 32 kbps. We assume each packet size is 30 bytes. The weight used in the exponential weighted moving average formula for packet service time (eq 3 in section 4.1) is set to 0.1. The epoch size is set to 100 ms and the maximum queue size is set as 30 packets. Throughout the simulation, we used a fixed workload that consists of 10 sources and 1 sink. The initial transmission rate was set to 4 pps (packets per second).

The IEEE 802.11 DCF has been used as MAC protocol and as a routing protocol we have used AODV in the simulation. We have run the simulation for 60 seconds.

We have compared our protocol with the recent rate control protocol DART [3] and SUPORTS [4]. Fig. 5 depicts the miss ratio under different offered load. We set the end-to-end deadline as 200 ms. It shows that as offered load increases the end-to-end miss ratio for our proposed scheme remains within 10% as compared to DART and SUPORTS and for their protocol it is much higher after a certain offered load. The miss ratio for multiple deadlines under different congestion for our proposed protocol is shown in Fig. 6. For three deadlines (100, 200 and 300 ms), it shows the higher the deadline the lower the miss ratio for several offered load. Here, only the packets with 100 ms deadline failed to maintain the miss ratio below 10%. So, for the application with much lower deadline the congestion detection threshold need to be changed. The end-to-end delay for different offered load is shown in Fig. 7. In this case, our proposed scheme maintains moderate E2E delay than the other two schemes. Fig. 8 illustrates the effective throughput for diverse offered load. The effective throughput is defined as the fraction of the packets received within deadline

over the load offered in the network. Our proposed scheme upholds about 80% effective throughput for different load. Due to the hop by hop adjustment of rate our proposed scheme and SUPORTS works well than sink initiated DART protocol.

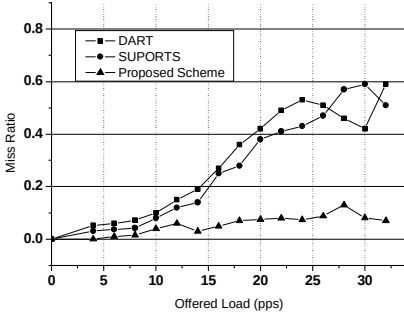


Fig. 5. Miss Ratio Under Different Offered Load

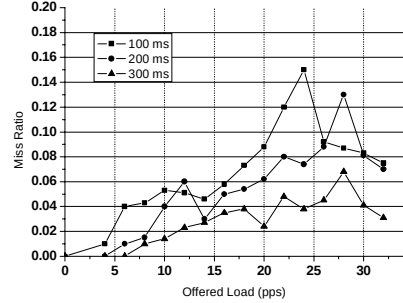


Fig. 6. Miss Ratio Under Multiple Deadlines

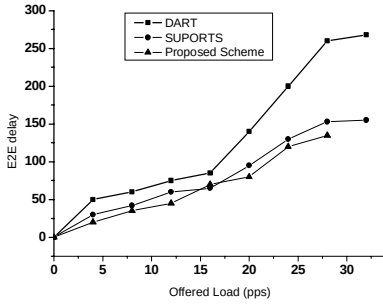


Fig. 7. E2E delay Under Different Offered Load

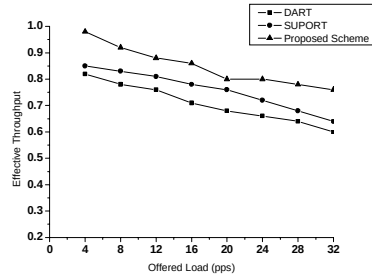


Fig. 8. Effective Throughput Under Different Offered Load

6 Conclusion

In this paper, we have presented an efficient QoS management scheme for providing better QoS under congestion situation through adjusting the traffic rate. We have demonstrated through the simulation that our proposed scheme achieves, i) Lower End-to-End Miss ratio, ii) A good Effective Throughput and iii) Lower End-to-end delay in diverse congestion situation. In future, our goal is to implement our protocol in real sensor test bed scenario.

References

1. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: Wireless sensor networks: A survey. *Computer Networks J.* 38, 393–422 (2002)
2. Wang, C., et al.: Upstream Congestion Control in Wireless Sensor Networks Through Cross-Layer Optimization. *IEEE Journal on Selected Areas in Communications* 25(4) (May 2007)
3. Gungor, V.C., Akan, O.B.: Delay Aware Reliable Transport in wireless sensor networks. *IEEE Journal of Communication Systems* 20, 1155–1177 (2007)
4. Kareons, K., Kalogeraki, V.: Real Time Traffic Management in Sensor Networks. In: *Proc of IEEE International Real-Time Systems Symposium* (2006)
5. Xia, F., Zhao, W., Sun, Y., Tian, Y.: Fuzzy Logic Control Based QoS Management in Wireless Sensor/Actuator Networks, *Sensors* 7, pp. 3179–3191 (December 2007)
6. Wan, C.-Y., Eisenman, S.B., Campbell, A.T.: CODA: Congestion Detection and Avoidance in Sensor Networks. In: *Proceedings of ACM SenSys.*, vol. 5-7, pp. 266–279 (November 2003)
7. Ee, C.T., Bajcsy, R.: Congestion Control and Fairness for Many-to-One Routing in Sensor Networks. In: *Proceedings of the ACM SenSys.*, vol. 3-5, pp. 148–161 (November 2004)
8. Hull, B., Jamieson, K., Balakrishnan, H.: Mitigating Congestion in Wireless Sensor Networks. In: *Proceedings of ACM SenSys.*, vol. 3-5, pp. 134–147 (November 2004)
9. He, T., Stankovic, J.A., Lu, C., Abdelzaher, T.F.: A spatiotemporal communication protocol for wireless sensor networks. *IEEE Trans. Parallel Distr. Syst.* 16(10), 995–1006 (2005)
10. Felemban, E., Lee, C.-G., Ekici, E.: MMSPEED: Multipath multi-SPEED protocol for QoS guarantee of reliability and timeliness in wireless sensor networks. *IEEE Trans. Mobile Comput.* 5(6), 738–754 (2006)
11. Akkaya, K., Younis, M.: An energy-aware QoS routing protocol for wireless sensor networks. In: *Proc. of Intl. Conf. on Distributed Computing Systems Workshops (ICSDSW)*, Washington, DC (2003)
12. Lu, C., Blum, B.M., Abdelzaher, T.F., Stankovic, J.A., He, T.: RAP: A real-time communication architecture for large-scale wireless sensor networks. In: *IEEE Real-Time and Embedded Technology and Applications Symposium*, San Jose, CA, pp. 55–66 (September 2002)
13. Network Simulator NS-2, <http://www.isi.edu/nsnam/ns>

Extended Concentric-Clustering Routing Scheme Adapted to Various Environments of Sensor Networks*

Jin-Young Choi, Sung-Min Jung, Young-Ju Han, and Tai-Myoung Chung

Internet Management Technology Laboratory,
Dept. of Electrical and Computer Engineering, Sungkyunkwan University,
300 Cheoncheon-dong, Jangan-gu,
Suwon-si, Gyeonggi-do, 440-746, Korea
{jychoi, smjung, yjhan, tmchung}@imt1.skku.ac.kr

Abstract. Cluster-based routing schemes are able to prolong the network lifetime and save the energy consumption in various environments of sensor networks. On the other hand, the chain-based routing protocols construct the chain such as PEGASIS and Concentric-Clustering Routing Scheme instead of a cluster formation. The concentric-clustering scheme is hybrid mechanism that combines chain-based scheme with clustering scheme. It divides sensor networks into some levels (like cluster), and one chain is assigned per level. However, main problem of this scheme is that the number of sensor nodes in the each level is unbalanced. This means that the unbalanced energy consumption among sensor nodes is occurred. Also, as this scheme selects the next cluster head to be closely located at current cluster head, energy of cluster head is dissipated quickly on the multipath among cluster heads. To overcome these problems, we propose the Extended Concentric-Clustering Routing Scheme Adapted to Various Environments of Sensor Networks. Our scheme fixes the number of sensor node on level for the same number of sensor nodes in each level. In addition, we consider the residual energy as the criterion of cluster head election for balanced energy consumption as well as the distance from lower cluster head or base station to upper cluster head. According to the simulation results, our scheme shows a better performance of energy consumption than other multiple-chaining schemes about 23%. Thus, our proposed scheme takes advantages of reducing energy consumption and prolonging the network lifetime.

1 Introduction

Wireless sensor network is a kind of the ad-hoc network that is composed of the number of sensor nodes. Sensor nodes are deployed on grid, random, and

* This research was supported by the MKE(Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Advancement) (IITA-2008-C1090-0801-0028).

concentration basis. Each sensor node transfers the data that is collected and monitored at the interested environment [1]. Sensor networks are designed to consider energy conservation, because sensor nodes have limited supply of energy, computational capacities, and small storage. To form energy-efficient network, many technologies have been developed such as routing protocols and data aggregation.

In the many routing protocols [2], the chain-based routing protocols such as PEGASIS [3], Concentric-Clustering Routing Scheme (CCRS) [4,5] are able to prolong network lifetime, as each sensor node except for the leader node only communicates with the closest sensor node through formed chain among sensor nodes. However, PEGASIS has a critical problem that the redundant transmission of the data is occurred. The reason of this problem is that PEGASIS does not consider the base station's location, when one of nodes is selected as the leader node. Moreover, PEGASIS is not suitable for various environments of sensor networks due to supporting only one-hop communication from a leader node to base station.

To improve the problem of PEGASIS, we proposed the Concentric-Clustering Routing Scheme (CCRS) which combines cluster-based protocol with chain-based protocol as a hybrid scheme [4,5]. We solve the problem of PEGASIS by forming multiple levels and having multiple cluster heads. The levels are formed into "concentric-clustering". It means that the shape of a cluster is concentric circles, when a wireless sensor networks are divided into several clusters. Specifically, this scheme partitions network into some clusters (called a level) according to strength of signal from base station and then constructs a chain by using greedy algorithm per cluster or level. After multi-chain is constructed, cluster heads are elected from each chain (or level, cluster), and each cluster head collects the aggregated data from its chain and transmits it by multi-hop to the base station. However, there are some problems. Firstly, the number of sensor nodes in each level is unbalanced. Secondly, energy of cluster head dissipates quickly on the multipath, because a sensor node as cluster head always selects the same neighbor sensor node as next cluster head, as this scheme selects the next cluster head to be closely located at current cluster head. Lastly, there is unbalanced energy consumption problem between sensor nodes, because it does not consider remaining energy of each node as the criterion of leader (called a cluster head) selection. Thus, this scheme is not suitable for various environments of sensor networks, because of unbalanced energy consumption.

In this paper, to solve the problem mentioned above, we propose the Extended Concentric-Clustering Routing Scheme Adapted to Various Environments of Sensor Networks (ECCRS). Our scheme fixes the number of sensor node on level for the same number of sensor nodes in each level. The number of sensor node on level is fixed to consider density. In addition, we consider the residual energy as the criterion of cluster head election for balanced energy consumption as well as the distance from lower cluster head or base station to upper cluster head. Therefore, even the sensor nodes are variously deployed on network,

energy of all sensor nodes are efficiently consumed. Our proposed scheme takes an advantage of improving the network lifetime.

The rest of the paper consists of the following. Section 2 reviews PEGASIS and Concentric-Clustering Routing Scheme, and section 3 describes Extended Concentric-Clustering Routing Scheme. In section 4, our proposed scheme is analyzed and compared with other schemes. In last section 5, we conclude this paper.

2 Related Works

In this section, we review PEGASIS and CCRS on basis of our scheme.

2.1 PEGASIS

PEGASIS is Power-Efficient Gathering in Sensor Information Systems. The main idea is for each node to receive from and transmit to close neighbors and take turns being the leader for transmission to the base station [3]. PEGASIS protocol operates as the follows.

A. Chain Construction

PEGASIS constructs the chain among sensor nodes by using a greedy algorithm which starts with the farthest node from the base station. One of entire nodes is selected randomly as a leader node.

B. Gathering and Transmitting Data

After whole network constructs chain, each node can know own next neighbor node on chain. To deliver the sensed data to the cluster head, each node receives data from previous neighbor and aggregates the data with own data. Next, the node transmits it to the next neighbor on the chain in every round.

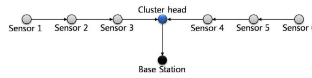


Fig. 1. Token passing approach of PEGASIS

This data gathering procedure is started at the two end nodes located at both sides of chain. The cluster head sends the token toward sensor 1, and then sensor 1 receives the token. Sensor 1 will pass own data toward cluster head through the next neighbor (sensor 2). Sensor 2 will pass the aggregated data that fuses own data with received data from sensor 1. Sensor 6 delivers the data along the chain from own to sensor 4 in the same way. Finally, a cluster head fuses own data with received data from sensor 3 and 4, and then transmits it to the base station.

2.2 Concentric-Clustering Routing Scheme (CCRS)

To improve PEGASIS, the concentric-clustering routing scheme is proposed.

A. Level Assignment

All sensor nodes are assigned to own level from the external base station. A level is given as the form like a concentric circle using the signal strength of the base station. At this point, the higher the number of level increases, the farther the distance between sensor nodes from base station is long. The number of levels is defined as $M_{all}^2 / \text{Size of Area}$ (M_{all} means the number of sensor nodes) [5]. For example, when 300 sensor nodes are deployed in the field of size $100m \times 100m$, the number of levels is defined as $300^2 / (100 \times 100) = 9$. The number of sensor nodes in each chain is different, but the distance between levels is same.

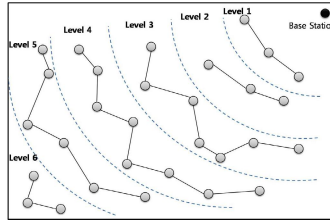


Fig. 2. Level assignment of CCRS

As shown in Fig. 2, level 4 has 8 sensor nodes, but level 1 has only 3 sensor nodes. If the number of sensor nodes in the level is small, it consumes the energy quickly, because sensor nodes in the level 1 is selected by $1/3$ probability.

B. Chain Construction in the Levels

Sensor nodes in each level construct a chain using the greedy algorithm as shown in Fig. 3. The external base station broadcasts data to sensor nodes in each level, after it performs the process of a chain construction. The chain construction is started at the farthest sensor node from base station [5].

C. Head Node Selection on the Chain

After chain is constructed, one of sensor nodes at each level is selected as cluster head as shown in Fig. 3. Cluster head in the highest level is simply selected by formulation like as $i \bmod M_L$ (M_L presents the number of nodes which has the same level) in i round [4,5]. Fig. 3 indicates that A, B, C, D, E, F nodes are selected as the cluster head.

D. Data Transmission

This scheme uses two data transmission. Firstly, in the intra-level communication, when the cluster head collects data in each round, it use a simple control token passing approach initiated by the cluster head to start the data transmission from the ends of the chain in its own level [4,5]. Secondly, in the inter-level communication, after the cluster head collects the data in its own level, the cluster head in each level delivers the data to cluster head in own lower level by multi-hop. After some interval, new round is started to select the new cluster head.

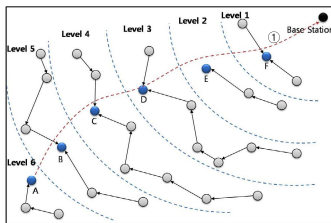


Fig. 3. Head Node Selection in CCRS

The ① line of Fig. 3 indicates that cluster heads are closely located at each other. This scheme takes an advantage of small transmission latency, but cluster heads on the ① line are quickly dead. If A is the cluster head at some round, B, C, D, E, F are always selected as other cluster heads.

3 Extended Concentric-Clustering Routing Scheme Adapted to Various Environments of Sensor Networks (ECCRS)

3.1 Assumptions

We propose the Extended Concentric-Clustering Routing Scheme adapted to Various Environments of Sensor Networks (ECCRS). Our scheme fixes the number of sensor node on each level for the same number of sensor nodes in each level. In addition, we consider the residual energy for the criterion of cluster head election for balanced energy consumption as well as the distance from lower cluster head or base station to upper cluster head. In this paper, our scheme has some assumptions as following:

- 1) The base station and all nodes are fixed in the given area.
- 2) The base station knows the location of all nodes, and each node has its unique ID.
- 3) The power of the base station is supplied infinitely.
- 4) The topology of networks is not changed.

3.2 Level Assignment

In this stage, our scheme is divided by several levels to deliver the aggregated data with multi-hop to the base station. Base station knows the number of cluster members. Author in [6] represents an optimal number of clusters in multi-hop in wireless sensor networks. The optimal numbers of clusters are between 10 and 20. For example, if all nodes are 300 and the number of cluster members is limited by 20, this network is divided by 15 levels. In other word, the number of sensor nodes in each level is same as the fixed value like as 20. As we statically fix the number of cluster members in each level, the sensor node that has the largest

amount of remaining energy and appropriate distance is selected as cluster head per round. Thus, we can guarantee balanced energy consumption among sensor nodes.

3.3 Chain Construction in Each Level

After level assignment, the farthest node from base station starts the chain construction to use the greedy algorithm. Each node receives the data that is the amount of remaining energy of close neighbor and calculates the average between own data and received data, and then node transmits the data to close neighbors. Therefore, we consider the residual energy of sensor nodes in the upper level as the criterion of cluster head election for balanced energy consumption. When the chain is constructed, each node calculates the average of remaining energy between itself and previous neighbor. Finally, cluster heads in each level calculate the average of remaining energy in own level. We will explain the detail of the calculation criterion in the next stage.

3.4 Selection of Cluster Head

After chain construction is completed, the base station selects the cluster head of level 1. At this time, the base station considers the remaining energy of each node and the distance from the base station to each node in the level 1. Sensor nodes in the level 1 transmit the own weight to the base station. The weight calculates the equation (1):

$$Weight = Energy_{oneself} / (Energy_{average} \times Distance) \quad (1)$$

The parameter means:

- 1) $Energy_{oneself}$ is the remaining energy of each node.
- 2) $Energy_{average}$ is the average of energy in the level.
- 3) $Distance$ denotes the euclidean distance between lower cluster head or base station and nodes of current level.

For example, if $Energy_{oneself}$ is greater than $Energy_{average}$ and Distance is relatively short between node and base station, the weight is high. The base station selects the cluster head that has the highest weight in the level 1. The selected cluster head broadcasts own location to be cluster head to sensor nodes of level 2. Continuously, cluster head of each level is selected by cluster head of previous level. When the cluster head is selected, it is important to consider the remaining energy of sensor nodes and the distance together between upper cluster head and lower cluster members. As a result, we can select energy-efficient cluster head on the chain.

3.5 Data Transmission

The process of data transmission is same as one of Concentric-Clustering Routing Scheme.

4 Performance Evaluation and Discussion

4.1 Radio Model

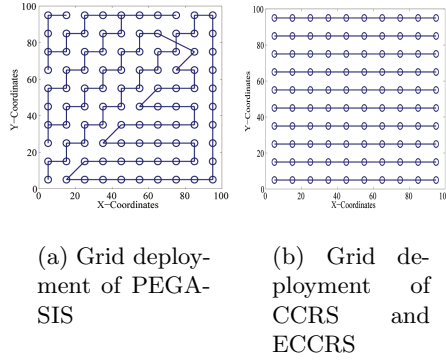
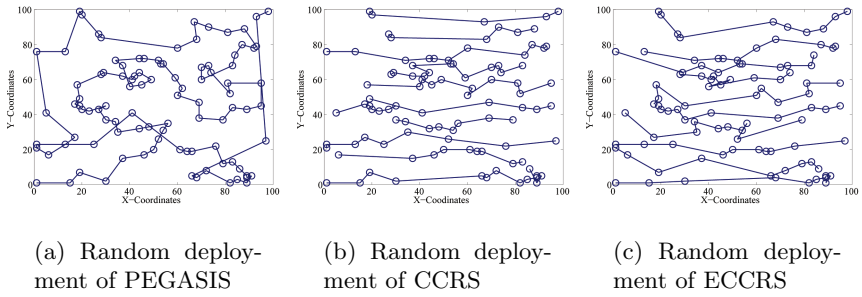
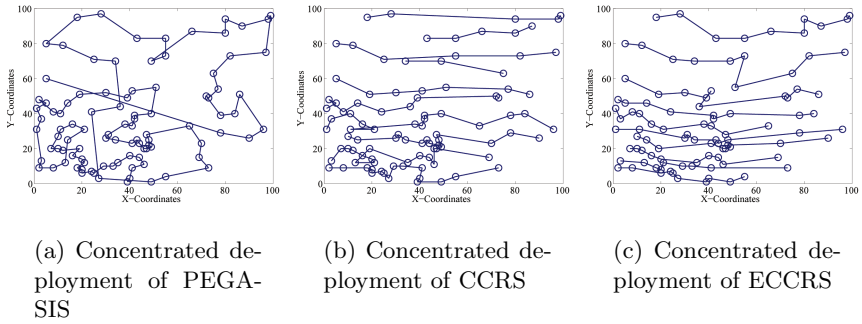
For analysis of the energy dissipation of the proposed scheme, we utilize first order radio model applied in [7,8] as a radio communication model among sensor nodes. In this model, the amount of energy consumption to transmit a packet with l -bit over distance d is given:

$$\begin{aligned} E_{TX}(l, d) &= l \times E_{elec} + l \times E_{fs} \times d^2, d < d_0 \\ &= l \times E_{elec} + l \times E_{mp} \times d^4, d \geq d_0 \end{aligned} \quad (2)$$

Where E_{elec} is the amount of energy consumption per bit in the transmitter or receiver circuitry, E_{mp} (for multi-path fading) and E_{fs} (for free space) are the amount of energy consumption in the RF amplifiers to counter propagation loss depending on the transmission distance d . The d_0 is determined as $\sqrt{\frac{E_{fs}}{E_{mp}}}$ by equating the two expressions at $d = d_0$. To receive a packet with k -bit, the radio expends $E_{rx}(k) = k \times E_{elec}$. In addition, the equation to aggregate k -bit each node is shown as $E_{Agg}(k) = k \times E_{Agg}$. The parameters of energy set up as $E_{elec} = 50\text{nJ/bit}$, $E_{Agg} = 5\text{nJ/bit}$, $E_{fs} = 10\text{pJ/bit/m}^2$, $E_{fs} = 10\text{pJ/bit/m}^2$, and $E_{mp} = 0.0015\text{pJ/bit/m}^4$.

4.2 Network Model

We consider grid, random, and concentrated deployment, because sensor networks may have various obstacles and environments. Although sensor networks are variously formed because of some obstacles or deployment regulation, they only consider random deployment. Therefore, we assume various sensor networks, where 100 sensor nodes will be deployed with grid, random, and concentration in the field of size $100m \times 100m$. The base station is located at (50, 200) outside the wireless sensor networks. Fig. 4, 5, and 6 show the chain constructions of PEGASIS, CCRS, and ECCRS at the various environments. In Fig. 4, CCRS and ECCRS use the same chain construction, but in Fig. 5, 6, each scheme forms different chain construction. Judging from the above figure, in (b) of Fig. 4, the energy consumption of CCRS and ECCRS is similar. In (b) of the Fig 5, sensor nodes of level 1 consume the energy quickly, because sensor nodes of level 1 have the 1/5 probability of selection as cluster head. If the number of sensor nodes in some level is big, the sensor nodes in this level consume the energy slowly. For example, sensor nodes of level 4, 6 dissipate the energy slowly, because they have the 1/13 probability of selection as cluster head. On the other side, in (c) of the Fig. 5, 6, each level has 10 sensor nodes as same. As the sensor network has equal density of each level, the lifetime of sensor nodes can be prolonged. For example, if the number of sensor nodes in each level is same, each sensor consumes the energy efficiently, because each sensor has the 1/10 probability of selection as cluster head from level 1 to level 10. Thus, network lifetime will prolong, and energy will be conserved.

**Fig. 4.** Comparing Grid Deployment of Each scheme**Fig. 5.** Comparing Random Deployment of Each scheme**Fig. 6.** Comparing Concentrated Deployment of Each scheme

4.3 Simulation Result

We used the MATLAB to evaluate the performance of PEGASIS, CCRS, and ECCRS. Fig. 7 shows the total remaining energy of whole networks in 2000 rounds. PEGASIS is not better than two other schemes. The total remaining energy of ECCRS is almost same as that of CCRS. In addition, we compared average and standard deviation of round of first dead node, because it is very important for comparing round of first dead node to evaluate the performance in sensor network.

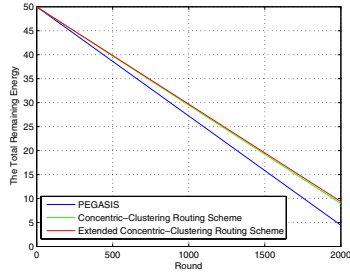


Fig. 7. The Total Remaining Energy of Each Scheme

Table 1. Average and standard deviation of round when first dead node is discovered

Scheme	Deployment	Average	Standard Deviation
PEGASIS	Grid	1240	-
CCRS	Grid	2040	-
ECCRS	Grid	2100	-
PEGASIS	Random	1256.2	392.5
CCRS	Random	1240.2	211.5
ECCRS	Random	1693	175.5
PEGASIS	Concentrated	1066	474
CCRS	Concentrated	1149	194.4
ECCRS	Concentrated	1619	156.4

According to Table 1, the average rounds of first dead node of PEGASIS and CCRS and ECCRS scheme are 1256.2, 1240.2, and 1693 rounds, when the algorithms are executed by 50 times with random deployment. This shows that the network lifetime of ECCRS is better than one of PEGASIS and CCRS. However, the standard deviation of ECCRS is smaller than one of PEGASIS and CCRS. Judging from Table 1, although the sensor nodes are variously deployed, we will expect the stable performance about network lifetime, because ECCRS considers density of level to be equal to density of other level. On the other hand, when the sensor nodes are deployed with concentration, the average rounds of first dead node of each scheme are 1066, 1149, and 1619 rounds. This result proves that performance of ECCRS is better than one of PEGASIS and CCRS.

5 Conclusion

In the various environments of wireless sensor networks, it is important to apply an energy-efficient cluster routing protocol. To be adapted to various environments, our scheme fixes the number of sensor node on level for the same number of sensor nodes in each level. In addition, we consider the residual energy as the criterion of cluster head election for balanced energy consumption as well as the distance from lower cluster head or base station to upper cluster head. Thus,

our scheme is a better performance of energy consumption than other schemes about 23%. In the future, we will research the optimized number of levels to efficiently consume the energy of all nodes and to improve the network lifetime for large-scale sensor networks.

References

1. Akyildiz, L.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A Survey on Sensor Networks. *IEEE Communications Magazine* 40(8), 102–114 (2002)
2. Al-Karaki, J.N., Kamal, A.E.: Routing Techniques in Wireless Sensor Networks: A survey. *IEEE Wireless Communications* 11(6), 6–28 (2004)
3. Lindsey, S., Raghavendra, C.S.: PEGASIS: Power-Efficient Gathering in Sensor Information System. In: *Proceedings of IEEE Aerospace Conference*, vol. 3, pp. 1125–1130 (2002)
4. Jung, S.-M., Han, Y.-J., Chung, T.-M.: The Concentric Clustering for Efficient Energy Consumption in PEGASIS. In: *International Conference on Advanced Communication Technology (ICACT)*, vol. 1, pp. 260–265 (February 2007)
5. Jung, S.-M., Han, Y.-J., Chung, T.-M.: The Concentric-Cluster Routing Scheme adapted to Large-Scale sensor Networks. In: *The 5th International Conference on Advances in Mobile Computing & Multimedia (MoMM)*, pp. 187–197 (December 2007)
6. Kim, N., Han, S., Kwon, W.H.: Optimizaing the Number of Clusters in Multi-Hop Wireless Sensor Networks. *IEICE trans. commun.* 1, 318–321 (2008)
7. Heinzelman, W.B., Chandarkasan, A.P., Balakrishnan, H.: Energy-Efficient Communication Protocol for Wireless Microsensor Networks. In: *Proceedings of the 33rd Hawaii International Conference on System Sciences*, Maui, HI (January 2000)
8. Heinzelman, W.B., Chandrakasan, A.P., Balakrishnan, H.: An application-Specific Protocol Architecture for Wireless Microsensor Networks. *IEEE Trans. Wireless Commun.* 1(4), 660–670 (2002)

Towards Cluster Based Wireless Sensor Network Deployment Management and Network Coverage Verification

Zhanyang Zhang

Computer Science Department
College of Staten Island, City University of New York
2800 Victory Boulevard,
Staten Island, NY 10314, USA
Zhangz@mail.csi.cuny.edu

Abstract. In a wireless sensor network, the problems of determining sensor locations and verifying their coverage after deployment are challenging ones and they are crucial for many wireless sensor applications during deployment and post-deployment assessment. In this paper, we propose a cluster based approach to address these problems using location guided laser beams as stimulating signals to trigger randomly deployed sensor nodes self-organizing into clusters within virtually defined geographical-grids. Each cluster has a cluster header that can communicate with a base station about the cluster location and its membership. Both cluster location and membership information provide valuable measures for sensor network operators to assess the degree of coverage and network connectivity. With a feedback radio link from the base station to the agent that is carrying out the deployment, it is possible for the deployment agent to correct any coverage problems during the deployment phase instead of costly post-deployment reactions.

Keywords: Wireless sensor networks, sensor clusters, k-coverage, and laser guide location Tracking.

1 Introduction

One of the key issues of wireless sensor network management is sensor node localization and assurances sufficient network coverage. Recently much research has been done to seek GPS-less solutions for locating sensors within a deployment region. Based on the published work so far, most of the solutions that provide sensor physical location data require a number of reference nodes with known physical locations and rely on communication of RF beacon signals. This approach uses signal strength and time of arrival (TOA) to estimate the physical sensor locations with considerable computational complexity and energy consumption [1, 2, 3]. Other research efforts addressing coverage problems have focused on network topology and connectivity [4, 5, 6, 7].

With today's technology, it is economical to pack multiple sensors into one wireless sensor unit that can sense many types of signals, such as, sound, light,

temperature, motion etc. The question motivating this research is whether we can use the output from sensors that respond to a sensible stimulating signal, such as a light beam projected a target location, to determine their physical locations? After all, it is most likely that sensors close to the target location will pick up the signal. Based on this observation, we take a novel approach to develop a stimulating signal based model to locate sensor clusters. Assuming that all sensor units have optical sensibility, our first choice of stimulating signal is a location guided laser beam, a technology widely used in military for targeting locations [8].

Our model is applicable to a wide range of sensor network applications that require large numbers of sensors randomly deployed to open fields, especially for monitoring hazardous and hostile environments such as a chemical spill or enemy controlled territory, where it is not possible to precisely deploy sensors by humans or human controlled vehicles. In such cases sensors are often deployed by airplane or artillery shells to the targeted areas. Obviously, the location scheme employed in this model requires an uninterrupted line-of-sight path for laser beams. However for many environmental monitoring applications, this is not a major problem.

In this paper, we take an integrated approach that can locate sensor clusters and verify the degree of coverage for a sensor network application based on the proposed model. First we present an algorithm that identifies a cluster of sensors by their geo-locations based on the stimulating signal in a predefined condition. Then we propose a cluster based feedback mode with a feedback radio link between the base station and a deployment/location tracking agent. Our model can provide wireless sensor network application operators both location and coverage information. We derive analysis results that show our model/algorithm scales well with the size of the sensor network (defined by the number of sensors and the size of sensing area).

We present the results of our simulation and performance studies to validate the model and its scalability measure as the sensor density and coverage areas increase using the NS2 Simulator [9].

2 Signal Stimulating Model (SSM)

In this model the sensor network is deployed and operated under optimal conditions defined by the set of assumptions.

1. Sensors – Each sensor has sensing, data storage, data processing and wireless communication capability that is equivalent to a MICA Mote (UC Berkeley) [10]. Each sensor covers a communication cell and a sensing cell defined by radius R_c and R_s respectively. All of the sensors are immobile and they can sense optical signals (delivered by laser beams).
2. Sensor network – A sensor network consists of a set of homogeneous sensors. Sensors can communicate with each other via wireless channels in single or multiple hops, thus they form an ad-hoc network. There are one or more base stations located outside the sensor region but near the boarder of the sensor network with wired or long range wireless communication links to the Internet for collecting data or disseminating queries and control commands to the sensor network.

3. Deployed region – Sensors are deployed over a flat area where there is an uninterrupted line-of-sight path for laser beams to reach ground level. A virtual grid marks this area. Each cell in the grid is a D -by- D square. To simplify the model, we assume there are no sensors located on the borders of the grids.
4. There is a lightweight location guided laser designator system that can project a laser beam to a given location (x, y) with acceptable accuracy [8]. In the case of no land access, a fly agent is need for sensor deployment.
5. To ensure the coverage and connectivity of a sensor network, the model requires that D , R_c and R_s satisfy the following conditions:
 - To ensure a sensor anywhere in a cell can cover the grid, it must satisfy the condition, $R_s^2 \geq 2D^2$. In our model we assume $R_s^2 = 2D^2$.
 - To ensure a sensor anywhere in a cell can communicate with a sensor anywhere in a neighboring cell, it must satisfy the condition, $R_c \geq 2R_s$. In our model, we assume $R_c = 2R_s$.

Fig. 1 below shows the parameters that define virtual grids with 4 neighboring grids.

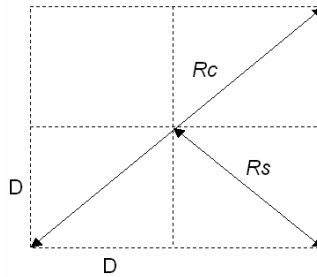


Fig. 1. Parameters that define virtual grids

A typical sensor network operates in five phases: the planning phase, deployment phase, post-deployment phase, operation phase and post-operation phase. In the planning phase, a site survey is conducted to evaluate the deployment environment conditions and to select a suitable deployment mechanism. A geographical virtual grid is developed as a result of the survey. In the deployment phase, sensors are randomly deployed over a target region. In the post-deployment phase, the sensor network operators need to identify or estimate the location of sensors and to assess the degree of coverage. A redeployment action might be needed to address significant blank spots and under-covered spots. The operation phase involves the normal operation of monitoring tasks where sensors generate data and communicate the result to the base station via multi-hops wireless links. The post-operation phase involves shutting down the sensor network by removing the sensors or setting them to sleep mode in order to preserve them for future operations.

Our model works during the deployment and post-deployment phase to identify sensor clusters (sensors that are located in the same cell of the virtual grid) and to estimate sensor cluster locations. It can verify the sensor network coverage using the

sensor cluster location data and the cluster membership data (the number of members in the cluster). Here we describe briefly how the model works:

Every point in the region can be represented by a pair of (x, y) coordinator values. A sensor has three possible states, U (unknown), H (cluster header) and M (cluster member). Initially all sensor states are set to U. During the post-deployment phase, an object flies over the deployed region and projects a laser beam to the center of a grid (X_c, Y_c). The sensors nearby will sense the signal. The sensor readings are stronger if they are closer to the projected laser beam. The sensor with the strongest reading is identified as cluster header (state=H). All the sensors that have a reading greater than λ (λ -cut) and are one hop away from the cluster header become members of the cluster (state=M). The ideal value for λ maximizes the possibility of including a sensor in the cluster if it is in the cell, and minimizes the possibility of including a sensor in the cluster if it is outside the cell. An optimal value of λ can be obtained through experiments and simulations. Fig.2 shows the cluster formed in grid 5.

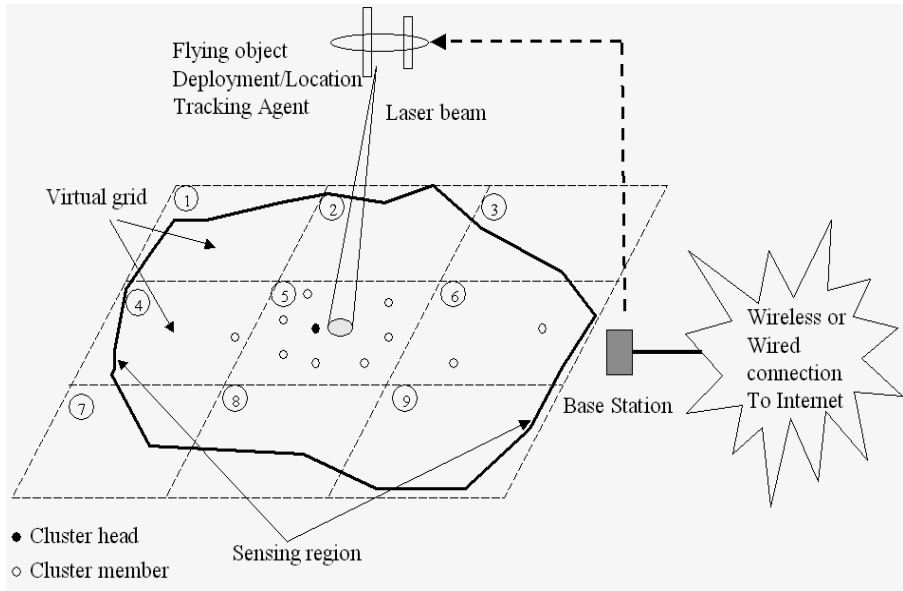


Fig. 2. A Cluster Based Feedback Model

In this paper, we assume that a laser beam is projected to one cell at a time with a cluster forming time interval, T , for each cell. T should be just long enough to allow the sensors in a cell to form a cluster, but not so long as to cause unnecessary delay for the operations between cells. In general, T is a function of sensor density - n (the number of sensor nodes in a cell), radio propagation delay τ and MAC layer back off delay β in the form of:

$$T = n^2\beta + n\tau + k \quad (1)$$

With this assumption, a sensor node can only belong to one cluster, since once it joins a cluster it will not respond to the stimulating signals projected to other cells.

3 Sensor Cluster Location Algorithm (SCLA)

Based on the SSM model, we present the formal algorithm with the following steps:

1. For each sensor with unknown status that has detected the signal, if the sensor reading is greater than λ , then it will broadcast a message with the sensor id and sensor reading (SID, Value) to its neighbors within one hop of communication. Otherwise it will keep silent.
2. When receiving a message, a sensor with unknown status acts according to the following rules:
 Rule1 - If the reading value of the received message is greater than its own reading and its own reading is greater than λ , then it will set the state=M (a cluster member).
 Rule2 - If a sensor does not receive a message and its own reading is greater than λ , or all of the received messages have reading values less than its own and its own reading is greater than λ , it will set the state=H (a cluster header).
3. The cluster header saves all the cluster members (sensor ids) in its local storage then sends its cluster location (grid location) and all members sensor id to the closest base station.
4. Repeat the above steps throughout all grid cells.

Based on the results of the SCLA algorithm, the base station knows which cell is covered and the degree of coverage (the number of sensors that can cover the cell). The base station can relay this information to the agent, who might currently be carrying out deployment and/or location tracking tasks. The agent can re-deploy a limited amount of sensors to a target grid in order to address any blank spots or under-coverage problems during the deployment phase. This will avoid the time delay of post-deployment assessment and costly re-deployment missions.

Our analysis shows the algorithm performs/scales well when sensor network size increases in terms of both the number of cells and the total number of sensors. Let L be the largest number of communication hops from a cluster header to the closest base station. Let M be the total number of cells. Let N be the total number of sensors deployed. Let $n(i)$ be the number of sensors in cell(i). The cost of the algorithm in terms of the number of messages communications is given as:

$$\text{Cost}(L, M, N) \leq \sum n(i) + M * L. \quad (i = 1 \text{ to } M, \text{ and } \sum n(i) = N) \quad (2)$$

If we assume sensors are uniformly distributed, then we have:

$$\text{Cost}(L, M, N) \leq M*(N/M) + M * L = N + M * L \quad (3)$$

It is equivalent to $O(N)$ when M and L is significantly smaller than N , which is true of most high-density sensor networks. Thus it is feasible to deploy large number of sensors in order to compensate for high rates of sensor node failure with limited overhead cost.

4 Simulation and Performance Evaluation

In order to validate the model and algorithm presented in this paper and to gain insights into how the algorithm works, we conducted simulation studies using the NS2 simulator with Monarch Extensions to ns [11].

Our simulations are implemented in two scenarios. The first scenario involves simulating a single cell grid with different sensor densities (number of sensors in the grid). The focus of this simulation is to study the performance and scalability of our model against sensor density. In the second scenario, we take the same measurements from a multi-cell grid simulation with considerations of both sensor density and the size of the deployment area in terms of the number of cells. The purpose of this simulation study is to understand the performance and scalability of our model in a multi-cell grid.

We set the cell dimension to 10 meters for all the simulations presented in this paper. Our simulation tests indicate that the outcomes are not as sensitive to the cell dimension as they are to sensor density. We let the number of sensor nodes be varied from 10 to 80 in increments of 10. In our simulation model, we set the propagation delay between two nodes as 10 ms ($\tau = 10\text{ms}$). We use multicast in UDP protocol to simulate sensor node broadcast in one hop distance. We set p as the probability for a node to receive the broadcast message successfully (p in the range 0 to 1). In our simulation we set $p=0.99$. The message package size is set to 128 bytes and the bandwidth between two nodes is set to 2mbs. To simulate IEEE 802.11 MAC layer data collision protocol, we introduce a back-off time delay, a random number between 50 and 100ms, which is assigned to a node when it detects that a channel is busy. The node will back-off for a delay interval before it tries to broadcast again.

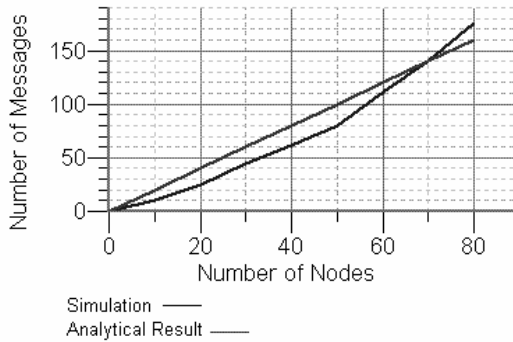


Fig. 3. Performance and scalability in a single cell measured in number of messages

The simulation results capture two key measurements, the number of messages being transmitted and the time interval for cluster formation in the grid. All of the simulation results represented below are the average of five simulation runs. Fig. 3 shows the number of messages being transmitted in a single cell grid. It compares the analytical result with simulation results. It indicates that the cost of message transmissions is close to a linear function of n , the number of sensor nodes in the cell.

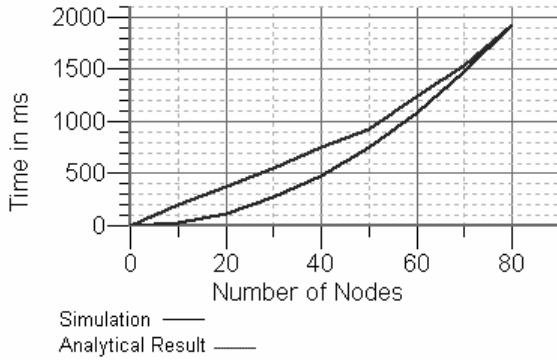


Fig. 4. Performance and scalability simulation in a single cell measured in time

Fig.4 shows the time intervals for cluster formation in a single cell grid. It compares the simulation result with the analytical results in (1). We set $\tau = 10$ ms and $\beta = 75$ ms, the average value of MAC layer back-off delay. To better understand cluster membership distributions and study the impact of λ values on member selections, we used a single cell grid of 10 meters by 10 meter with λ values in the range $[0.02, 0.04]$. The simulation result in Fig. 5 shows the percentage of sensors that are dropped from the cluster for the cell as the value of λ changes. It shows that higher λ values result in more sensor nodes being excluded from the cluster.

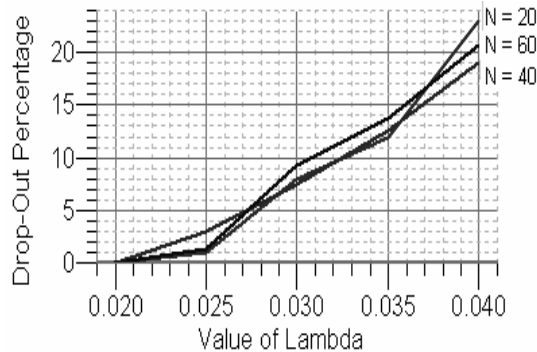


Fig. 5. Membership distribution with the impact of λ

The simulation results below are for a multi-cell grid scenario with the same key measurements as we presented for a single cell grid. Fig. 6 shows that with a fixed number of sensor nodes, the number of messages being transmitted actually drops as the number of cells increase. This is not exactly what we expected from the cost function defined in (3). This simulation result indicates that the number of messages being sent is more sensitive to the density in each cell than the number of cells in a grid.

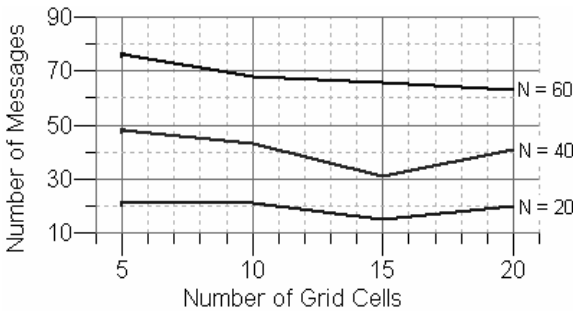


Fig. 6. Performance and scalability simulation in a multi-cell grid measured in messages transmitted

Fig. 7 indicates that the probability of data collision at the MAC layer decreases as sensor density decreases. For this reason, we see a slower growth in time in a multi-cell grid with a fixed number of sensors in comparison with Fig. 4.

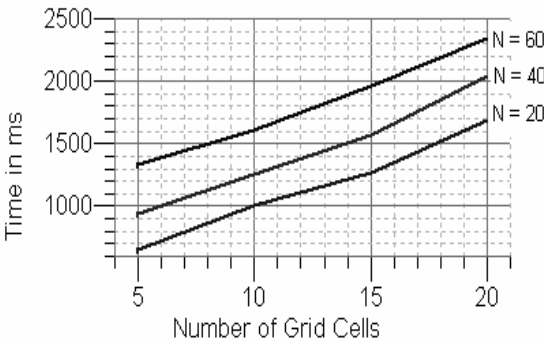


Fig. 7. Performance and scalability simulation in a multi-cell grid measured in time

Fig. 8 presents an interesting measurement, the percentage of sensors wrongly claimed by clusters. It is closely correlated with the λ values. The ratio of disputed sensors drops or stays at the level after the number of cells reaches 16 due to the decrease in sensor density.

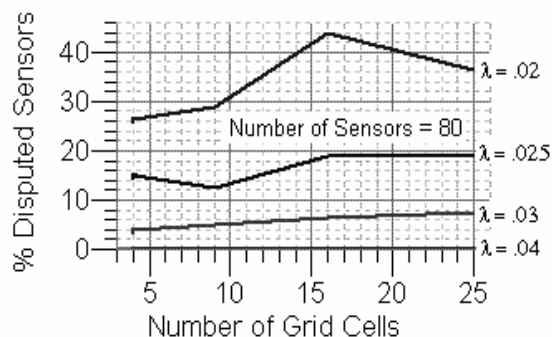


Fig. 8. Number of disputed sensors with λ values

Fig. 9 shows the percentage of 80 sensors uniformly distributed over a multi-cell that remain unclaimed for a range of lambda values.

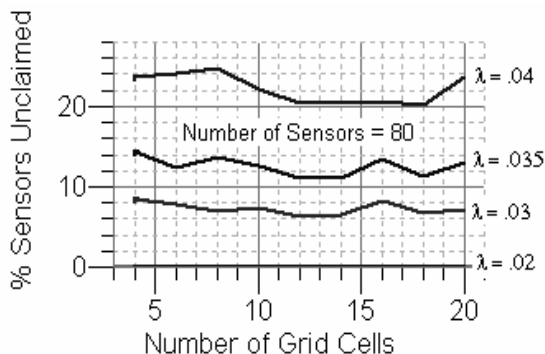


Fig. 9. Percentage of unclaimed sensors with λ values

5 Conclusion

Our model can identify sensor clusters and estimate their proximate locations. These results are significant for many types of sensor networks applications. We have identified three areas for which the geo-location data of sensor clusters is extremely useful; determination of sensor network coverage, self-organization of sensor clusters for energy efficient operations and to prolong the life of a wireless sensor network, and location based ad-hoc routing protocols in sensor networks. For future research, we plan to introduce a fault-tolerant model, and to test the robustness of the model by considering different sensor failure scenarios and deployment terrains. We will also compare the performance of our model with some previously reported solutions in terms of computational complexity and energy consumption.

Acknowledgement. This work was supported (in part) by a grant from The City University of New York PSC-CUNY Research Award Program (No. 60085-3435), a grant from CUNY Research Foundation (80210-14-04) and the funding from the New York State Dormitory Authority (GS7022).

References

1. Akyildiz, I.F., et al.: A Survey on Sensor Networks. *IEEE Communications Magazine* (2002)
2. Langendoen, K., Reijers, N.: Distributed localization in wireless sensor networks: a quantitative comparison. *Computer Networks* 43(4), 499–518 (2003)
3. Patwari, N., et al.: Locating the nodes: cooperative localization in wireless sensor networks. *IEEE Signal Processing Magazine* 22(4), 54–69 (2005)
4. Wang, Y.: *Topology Control for Wireless Sensor Networks*, London. Book *Wireless Sensor Networks and Applications*, pp. 113–147. Springer, Heidelberg (2007)
5. Zhang, H., Hou, J.C.: Maintaining Sensing Coverage and Connectivity in Large Sensor Networks. *Ad Hoc & Sensor Wireless Networks*, vol. 1, pp. 89–124. Old City Publishing Group (2007)
6. Huang, C.-F., Tseng, Y.-C.: The Coverage Problem in a Wireless Sensor Network. *Journal Mobile Networks and Applications* 10(4), 519–528 (2005)
7. Wang, X., et al.: Integrated Coverage and Connectivity Configuration in Wireless Sensor Networks. In: *Proceeding of ACM SenSys. 2003*, Los Angeles (2003)
8. Defense daily networks – the business source for aerospace and defense: Lightweight Laser Designator Rangefinder, <http://www.defensedaily.com/progprof/army/llldr.pdf>
9. The Network Simulator NS-2 Documentation (2008), <http://www.isi.edu/nsnam/ns/ns-documentation.html>
10. Crossbow Inc. Product Specification: MICA Wireless Measurement System (2006), http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICA.pdf
11. Monarch Project: Monarch Extensions to NS2, <http://www.monarch.cs.cmu.edu/>
12. Broch, J., et al.: A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. In: *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, Dallas, Texas, USA (1998)

A Logical Group Formation and Management Mechanism Using RSSI for Wireless Sensor Networks^{*}

Jihyuk Heo, Jin Ho Kim, and Choong Seon Hong^{**}

Department of Computer Engineering, Kyung Hee University,
Socheon-ri, Giheung-eup, Yongin-si, Gyeonggi-do, 446-701, South Korea
jihyuk@khu.ac.kr, jhkim@networking.khu.ac.kr,
cshong@khu.ac.kr

Abstract. Wireless sensor network is a suitable technology for ubiquitous environment. However, in WSN, as the network size grows larger, overheads such as flooding, calculation and memory overhead rapidly increase as well. For reducing the overhead of Micro Controller Unit (MCU) calculation and energy consumption, partitioning of WSN in several groups and hierarchy is suggested. For group formation, these mechanisms typically utilize predesigned network group information or location aware algorithms. However, the existing methods which use network topology information have node distribution and scalability problem in dynamic networks. Also, methods which use location-awareness have energy consumption problems. Therefore, we suggest a logical group formation and management mechanism using sensor association information and received signal strength indication without using any additional devices. With the help of proposed mechanism, WSNs would have stability and long lifetime.

Keywords: Wireless sensor networks, WSN, logical group formation, group management, RSSI.

1 Introduction

Wireless Sensor Networks (WSNs) are organized with a number of tiny sensor nodes. Unlike personal computers, sensor nodes are typically characterized by their limited computing power, limited memory, low power resources and expectedly low cost. Diminutive in size, easy deployment and low cost make sensor nodes a perfect candidate to build a Ubiquitous computing environment.

Because, WSNs are usually deployed to sense certain phenomenon and then forwarding the sensed data to the sink, numbers of small size packets are flowed. However, sensor nodes have limited resources. As a result, large scale WSN has less stability. Similarly, organizing and managing groups are also needed for dynamic network stability.

^{*} "This research was supported by the MKE under the ITRC support program supervised by the IITA" (IITA-2008-(C1090-0801-0002))".

^{**} Corresponding author.

In this paper, we propose a logical group formation and management mechanism for constructing proper number and size of groups using sensor node association information and Received Signal Strength Indicator (RSSI) without additional devices. Through proposed mechanism, WSNs would have stability and long lifetime.

The rest of this paper is organized as follows. In section 2, related work in the field is discussed. In section 3, the proposed logical group formation and management mechanism is defined. In section 4, the performance of the proposed mechanism is analyzed against other group formation scheme and scheme using ad-hoc location aware system. Finally, in section 5, we conclude our work and discuss future work.

2 Related Works

As the WSN size grows bigger and bigger, various overheads increases as follow.

Flooding overhead is first. Because sensor nodes enter or leave the network frequently, the WSNs are very dynamic. So, proactive (on-demand) routing protocol is used for routing algorithms. For example, in ZigBee specification [2], AODV (Ad-hoc On-demand Distance Vector) is used. To discover route to destination, routing control messages are flooded to all network nodes. This flooding is big overhead to sensor node which has limited resources.

Next, decrease of network lifetime. Because of the nature of broadcast, it causes duplicated transmission and broadcast handling cost is very high. It consumes much energy and makes network lifetime shorter consequently.

Finally, as the WSN size increases the ‘key management’ becomes more difficult and the probability of a node of being compromised or key leakage increases as well. Group based network can handle these problems even if size of the network is large. Therefore, group based network management is needed. There are 3 kind way to group formation as follow.

First method uses group topology information. In this method, sensor nodes are deployed on predefined places. Furthermore, nodes are also assigned their group before installation. This method should draw a plan about network size, the number of groups, group deployment position and so on. This method, as it seems, is too strict to adapt dynamic network environment.

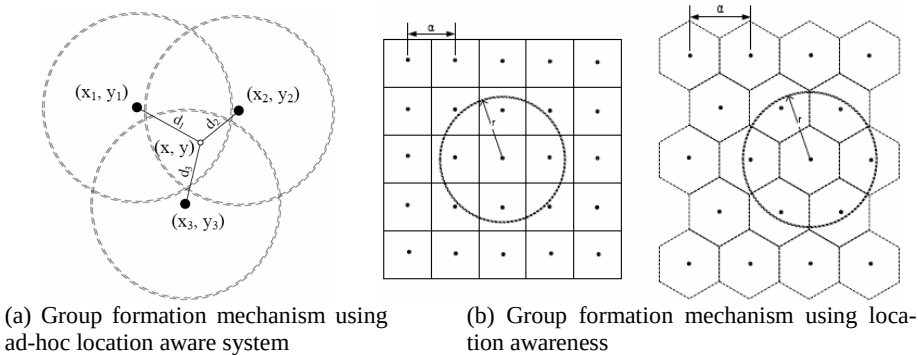


Fig. 1. Group formation mechanism

Second method uses additional device or ad-hoc location aware system for group formation. Firstly additional device like Global Positioning System (GPS) is attached to sensor node for measuring absolute location [3]. Secondly, other sensor nodes calculate relative positions from location aware sensor nodes [4][5]. As you can see in (a) of figure 1, location aware sensor node knows its own location and spread beacon with its own location information. So, sensor nodes can calculate relative position. Through this algorithm, WSN designs rectangle or hexagon shape group like (b) of figure 1. However, the price of GPS chip is much expensive than sensor node [8] and the power of location aware sensor nodes exhaust drastically due to spreading location information which makes this solution less appealing.

Last method makes group logically. WSNs are expanded through node associations. In this association procedure, the new incoming sensor node makes relationship with neighbor node [1][2]. Hence, the nodes which have relationship are located closely. This method uses that association information for group formation [6].

Through one of the above 3 discussed methods, WSN operates in two tiers and operated hierarchically. In the lower level, broadcast spread only within own group. When a sensor node of one group wants to transmit to the sensor node of the other group, data transmission is performed via upper level.

In recent research, location aware algorithm is popularly used for group formation and management. But, regularly used GPS is very expensive than sensor node which has a purpose to reduce the product cost to 2 dollars [8]. Because location awareness is used during initialization phase only, method using location aware algorithm is expensive solution for WSN as most of the time expensive devices are not used. Therefore, mechanism for logical group formation and management without additional device like GPS or ad-hoc location aware system needs to be researched.

3 Proposed Scheme

When a new incoming node joins the network, that node chooses one of the existing node as a parent which has lowest depth (depth means location level of tree structure and it started from root as zero) and highest Link Quality Indicator (LQI; RSSI is a kind of LQI) within neighbor nodes [1][2].

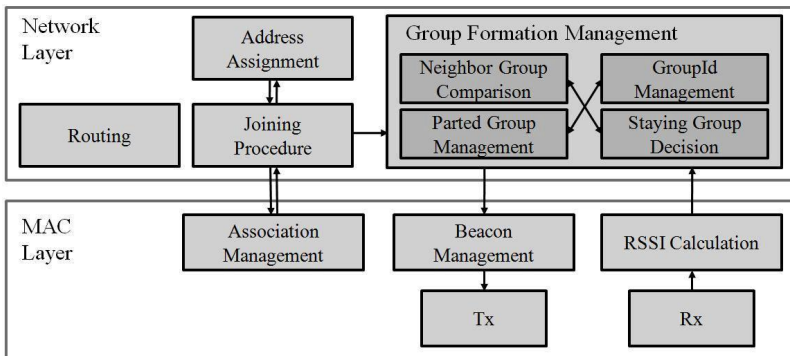


Fig. 2. Group Formation Management Architecture

As you can see in figure 2, our proposed mechanism for logical group formation and management uses association information and calculated RSSI through transmitted and received packets.

3.1 GroupId

Each group has identity which is called GroupId. The GroupId could be described as two ways. Like figure 3, 16/32 bits GroupId is divided into various bits for describing group hierarchy.

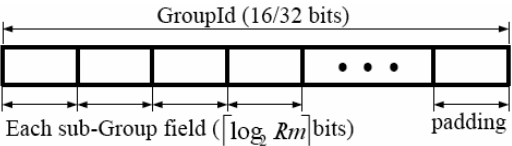
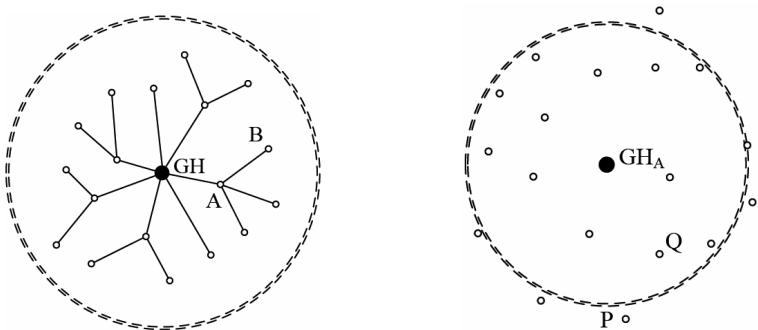


Fig. 3. GroupId for describe group hierarchy

In ZigBee specification, maximum number of router (R_m) is defined. The R_m is available router number for a sensor node to associate; default value of R_m is 0x06. According to figure 3, each sub group has 3 bits ($= \lceil \log_2 6 \rceil$ bits) to describe group hierarchy. Through this described method, subnetting/supernetting algorithm could be applied to group. However, it could not be flexible to describe group hierarchy of too dynamic network because group hierarchy structure too easy to break when sensor nodes frequently enter and leave the network. So, this method is used for relevantly stable network environment.

On the other hand, The GroupId can be described just by 16bits short address of header nodes. Only 16 bits address space is needed to describe GroupId and it has flexibility to describe dynamic network. However, this case is not able to describe group hierarchy. Therefore, this method is used for dynamic network environment.



(a) Group expansion according to sensor (b) Construction of parted group nodes association

Fig. 4. Group construction and expansion

3.2 Expansion of Group

When new incoming sensor node enters the network, it tries to become a part of its parent's group. From this algorithm, group is expanded later on. In this procedure, nodes spread beacon to give their own information to neighbor nodes. Through the information contained in a beacon nodes can recognize neighbor group headers.

Figure 4 (a) shows that wireless sensor network which is organized as tree structure. The connected lines mean parent-child relationship of nodes. The entered sensor node tries to belong to group G as long as RSSI, measured with group header GH, is higher than threshold.

3.3 Parted Group Generation

If sensor node is starting point of network like sink or if RSSIs which are measured with neighbor group headers are below than threshold (it means there is no near group header), sensor node cannot able to belong to the group according to algorithm described in 3.2, Expansion of group. In these cases, sensor node construct parted group.

$$P_r = \left(\frac{K}{4\pi R} \right)^2 \cdot G_t \cdot G_r \cdot P_t$$

$$R = \frac{K}{4\pi} \sqrt{\frac{G_t \cdot G_r \cdot P_t}{P_r}} \Rightarrow R \propto \frac{k}{\sqrt{P_r}}$$

P_r : Receiver side electric power [w] P_t : Sender side electric power [w]

K : Used wavelength (c/f) [m]

R : Distance between sender and receiver [m]

G_r : Receiver side antenna electric power gain [dB]

G_t : Sender side antenna electric power gain [dB]

Fig. 5. Received Signal Strength Indicator (RSSI) Formula

To calculate distance between sensor node and group header node, we can use RSSI. All sensor node can calculate RSSI by formula shown in figure 5 when sensor node transmits or receives packet [1]. The sensor node can guess distance between two nodes according to the formula, because all parameters except R and P_r are fixed in hardware specification. Therefore, new incoming node decides whether belongs to group of parent node or constructs a parted group according to RSSI threshold. For being separate group, parted group should operate merge procedure with nearby groups.

Figure 4 (b) shows the example of parted group construction. Sensor node A and B are belong to group G_A because RSSI of nearest group header GH_A is strong. However, Sensor node P constructs new parted group, because RSSIs of neighbor group header are weak or do not exist.

3.4 Merge of Parted Group

In boundary of group, there are many parted groups which do not belong to the ordinary group. Because sensor nodes which are located in group boundary have low signal strength with ordinary group header, that sensor nodes try to construct parted groups respectively. Therefore, we need merge procedure for a number of parted groups.

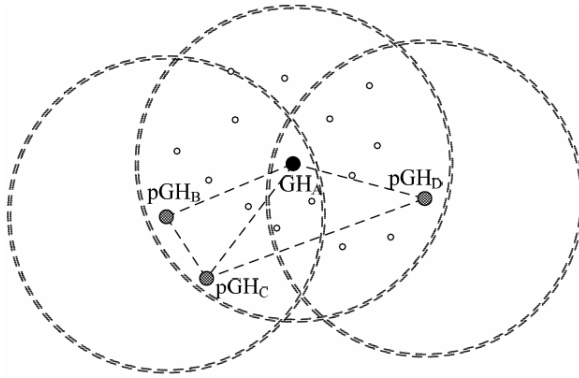


Fig. 6. Merge procedure of parted groups

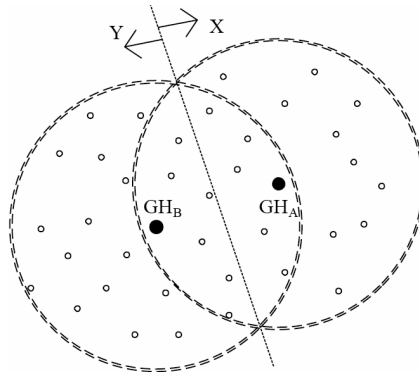


Fig. 7. Decision of normal sensor node to belong to group

Parted group header calculates distances with neighbor group headers or neighbor parted group headers. Then, merge procedure is performed between close distanced group headers. And, group header which has low depth and smaller address going to be a group header of merged group. If parted group header is isolated, it becomes a separated group.

Figure 6 shows that merge procedure of parted groups and group construction of parted group. Node pGH_B , pGH_C and pGH_D are parted group headers, because they have long distance with group header GH_A . Parted group header node pGH_D has no suitable group to merge, so the parted group header pGH_D going to be separate group header. However, parted group header pGH_B and pGH_C are located closely, so they perform merge procedure. Parted group header pGH_B and pGH_C have same depth. Because parted group header pGH_B has lower address, parted group G_C merge to parted group G_B .

3.5 Belonging to Group Decision of Sensor Node

New group formation has an effect on near sensor nodes. Normal sensor nodes belong to group for efficient energy consumption. Normal sensor nodes monitor the distance with group header nodes and try to belong to nearest group.

Figure 7 shows that after node GH_B separates from group G_A , node GH_B constructs group G_B . As you see in figure 7, the overlap region is made by two group header nodes GH_A and GH_B . Sensor nodes which are located in this overlap region can decide which group to belong according to comparison of RSSI. Usually, sensor nodes which are located X direction of dotted line belong to group G_A and sensor nodes which are located Y direction of dotted line belong to group G_B .

3.6 Group Reformation and Management

In the heterogeneous network, sensor node which has routing capability and relevantly more battery power takes a role of group header. Normal sensor nodes transmit packet via group header. So, group header node consumes more energy. As a consequence, group header replacing mechanisms are suggested in heterogeneous network. In this paper, group is formed by radio range of group header node, so group header replacing mechanisms have an effect on sensor nodes group association.

After group header node is changed according to group header election algorithms, normal sensor nodes should be reconfigured. When the group header node is changed, normal sensor nodes which are close or distant from group header node are reconfigured by the algorithm explained in Section 3.5.

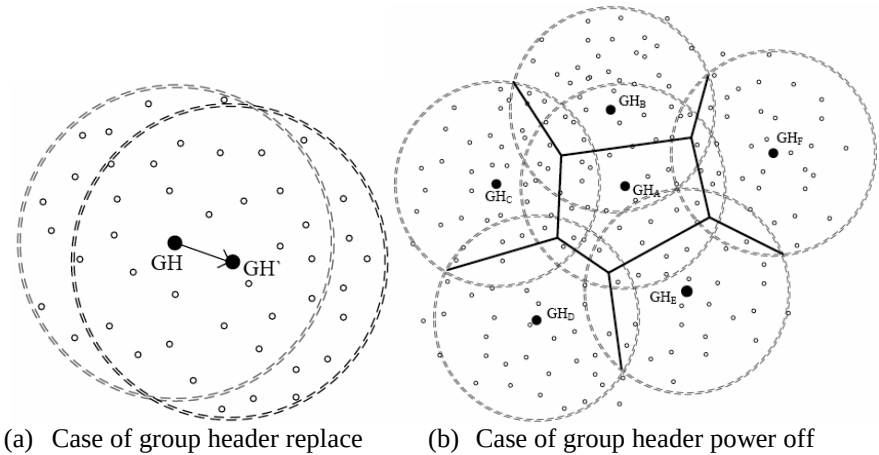


Fig. 8. Reconfiguration of group formation

Figure 8 (a) shows that sensor node reconfiguration when the group header is changed. Right bottom sensor nodes which are close to group header node GH' belong to group G' and left top sensor nodes which are far away from group header node GH' belong to other group or construct a new parted group.

Group header node disappearance is another reason of group reconfiguration. When the group header node performs power off or suddenly break down, sensor nodes which are belonged to that group, have to find new group to belong. Some sensor nodes which are located at the boundary belong to nearer groups. Other sensor node which are located the center construct new parted group and perform merge and belong procedure according to the description given in Section 3.4. and 3.5.

Figure 8 (b) shows that example of WSN with 6 groups. In this case, if group header node GH_A of center group is down, sensor nodes which are overlapped with other group G_B, G_C, G_D, G_E, G_F respectively belong to group G_B, G_C, G_D, G_E, G_F . And, sensor nodes which are located at the center of group G_A respectively construct parted group and merge.

Finally, we can form and manage logical group using node association information and RSSI without location aware algorithms or additional chip.

4 Simulation Results

To evaluate the performance of our proposed scheme we have performed simulations using NS-2 on Fedora Core 3. The IEEE 802.15.4 Medium Access Control Protocol is employed as the Data Link Layer. In the simulations, sensor nodes are deployed randomly.

In simulation, we compare our proposed mechanism with a logical group formation scheme using association information [6] and Centroid [5] using location aware algorithm. And we can evaluate efficiency of our proposed mechanism through measurement of group completing time and the number of groups according to RSSI threshold.

Figure 9 shows the group formation completing time with different number of sensor nodes. The Group formation completing time means time required to assign every

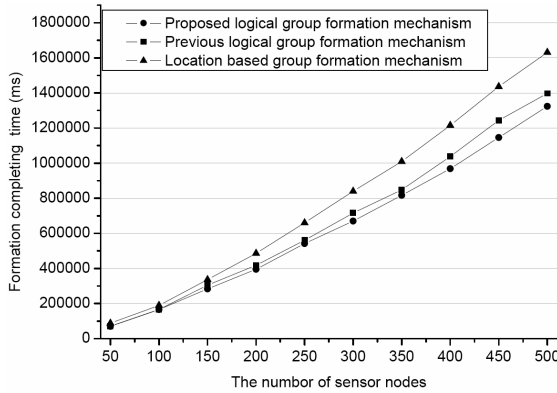


Fig. 9. The group formation completing time

sensor node a group. In each case i.e. proposed mechanism, a logical group formation scheme [6] and Centroid [5], the number of sensor nodes is changed by 50, 100, 150, 200, 250, 300, 350, 400, 450 and 500.

In figure 9, we can see that group formation completing times of logical group formation mechanisms are similar. However, group formation completing times of logical group formation mechanisms are shorter than the location aware group formation mechanism.

The logical group formation mechanisms just calculate and compare without additional RF transmission because nodes decide belonging just according to logical

information. However, the mechanisms which use location aware algorithms use transmission of location beacon for confirming its own location. The Additional RF transmission brings time delay and energy consumption.

Figure 10 shows that the number of formed groups according to threshold of RSSI with randomly deployed 100 sensor nodes in 50 square meter.

For convenience, we indicate RSSI by integer value from 0 to 255. Thresholds of RSSI are set 128, 153, 179, 204, 230 and 255.

As figure 10 shows, when the threshold of RSSI is set to 204, the number of formed groups is 23 and the average number of sensor nodes belong to a group is approximately equals to 4. Also, at that point, the energy is consumed less than other.

From above simulations, proposed logical group formation and group management mechanism is more efficient in time and cost of network structure than existing mechanisms that use location aware algorithms [5] or logical group formation scheme [6].

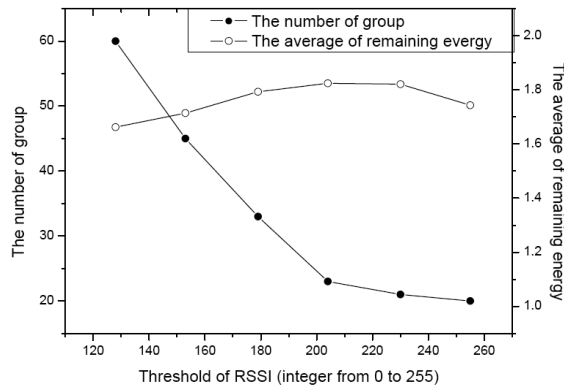


Fig. 10. The number of formed groups

5 Conclusion

In the paper, we suggest a logical group formation and group management mechanism. Using proposed mechanism, WSNs could make and manage proper number and size of groups without additional devices.

Our proposed mechanism uses association information between sensor nodes and RSSI value. Our algorithm operates in following 3 phases for group formation and management. At first, according to RSSI value among group header nodes of own group and neighbor group header nodes, sensor nodes determine whether to make a new group or not. Next, normal sensor nodes decide which group they belong to according to RSSI value of neighbor group header nodes. Finally, group header nodes compare RSSI value of neighbor group header nodes and determine whether to merge group or not.

Effects of sensor node density to logical group formation and management and verification of proposed mechanism through implementation on test-bed are left as the future work.

References

1. IEEE Std. 802.15.4-2006, IEEE Standard for Information technology — Telecommunications and information exchange between systems — Local and metropolitan area networks — Specific requirements - Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs) (2006)
2. ZigBee Standards, ZigBee Document 053474r15, ZigBee Alliance (February 2007)
3. Luo, H., Ye, F., Cheng, J., Lu, S., Zhang, L.: TTDD: Two-Tier Data Dissemination in Large-Scale Wireless Sensor Networks. *Wireless Networks* 11(1-2) (February 2005)
4. Chu, H.-C., Jan, R.-H.: A GPS-less, outdoor, self-positioning method for wireless sensor networks. *Ad Hoc Network* 5(5) (July 2007)
5. Bulusu, N., Heidemann, J., Estrin, D.: GPS-less Low Cost Outdoor Localization for Very Small Devices. *IEEE Personal Communications Magazine* (October 2000)
6. Lee, J.W., Heo, J., Hong, C.S.: A Logical Group Formation and Key Distribution Scheme in WSN. *Journal of Korea Institute of Information Scientists and Engineers* 34(4), 296–304 (2007) (in Korean)
7. Miura, H., Sakamoto, J., Matsuda, N., Taki, H., Abe, N., Hori, S.: Adequate RSSI Determination Method by Making Use of SVM for Indoor Localization. In: Gabrys, B., Howlett, R.J., Jain, L.C. (eds.) *KES 2006. LNCS*, vol. 4252, pp. 628–636. Springer, Heidelberg (2006)
8. Digital Times,
http://www.dt.co.kr/contents.html?article_no=2007011702011757730004 (in Korean)

OD Count Estimation Based on Link Count Data^{*}

Yi Jin¹, Dongchen Jiang¹, Shuai Yuan¹,
Jianting Cao¹, Lili Wang², and Gang Zhou¹

¹ State Key Lab. of Software Development Environment,
Beihang University, Beijing, China, 100083
² School of Computer Science and Engineering,
Beihang University, Beijing, China, 100083

Abstract. TM (Traffic Matrix) estimation is a hot research area recently. Current TM estimation methods are generally designed for backbone and ISP networks. They estimate complete TM which is unnecessary for many IP networks in reality and especially unsuitable for the networks that have many entries. In this paper, we propose an estimation algorithm that is designed for IP networks on link layer. Our algorithm estimates the OD (Origin and Destination pair) count on the basis of link counts which are easy to obtain. Our algorithm first builds a three-entry virtual network from actual network, and then achieves the final result by multivariate linear regression. We verify our algorithm in the official network of our lab by comparing with exact OD count data that are obtained by NetFlow.

Keywords: Traffic Matrix, OD Count, Link Count, Multivariate Linear Regression.

1 Introduction

Traffic matrix (TM) describes the volumes of traffic that flows between every possible OD pair (the pair of Origin and Destination). Compared with common network traffic monitoring, TM is more detailed to reflect the actual network traffic and widely used in capacity planning, traffic engineering, reliability analysis, network management etc. TM estimation is an effective methodology to obtain TM, based on indirect measurement data that are easy to collect.

Most existing TM estimation algorithms are designed for backbone and ISP networks so that they are unsuitable for networks having many entries which widely exist in reality. Because TM is a matrix of OD counts whose amount is direct proportional to the square of network entries. And almost all current algorithms estimate the complete TM which means huge computation in the case of many entries. However, in actual applications, complete TM is usually

^{*} This paper is supported by National Basic Research Program of China (973 Program), No.2005CB321901.

unnecessary and obtaining required OD counts (part TM) is sufficient enough. Therefore, in this paper we propose an estimation algorithm that only estimates designated OD counts. This algorithm is on IP network link layer, based on link count data which are easy to collect.

Our algorithm is composed of two steps: first map complex actual network into an equivalent three-entry virtual network which makes the next estimation simple and independent of various network topologies. The second step is to estimate designated OD counts based on link count data by multivariate linear regression. In this step, we choose an empirical network traffic mean-variance relationship as the basic estimation assumption. Our algorithm is verified in our laboratory's official network by comparing with the exact OD count data measured by NetFlow.

The rest of this paper is organized as follows. The related work is discussed in section 2. And the whole estimation algorithm is provided in section 3. In section 4 we apply our algorithm in our laboratory's official network and verify the estimation result. Finally, in section 5 we conclude the whole paper.

2 Related Work

2.1 Direct Measurement vs. Indirect Measurement

There are two mechanisms to obtain TM data: the mechanism based on direct measurement and that based on indirect measurement. The former gets TM mainly on basis of Flow techniques (NetFlow[10][2], sFlow[11] etc.) which analyze packet content at different protocol layers. Thus exact TM data could be obtained. However, Flow techniques are not widely supported yet in actual networks, i.e. it is generally hard to get TM data through direct measurement in reality. The second mechanism statistically estimates TM data based on devices' statistical data, such as link count. Though this mechanism is not as accurate as the former, its original data are generally very easy to collect. Therefore, it could be easy to apply in actual networks.

In our paper, we use link count, which could be easily collected from all network devices. And we choose SNMP to collect link count data, because it is widely supported by almost all network devices.

2.2 Mean-Variance Relationship

The traffic of one network entry is a linearly aggregation of OD pairs. If a network has n entries, there should be $n \times (n-1)$ OD pairs, i.e. the linear equation between unknown OD counts and measurable link count is always indeterminate. Therefore, we adopt a mean-variance relationship of network traffic as basic statistical assumption of our estimation, which makes the second order statistics of link counts available for the estimation.

Up to the present, the mean-variance relationship is still obtained through empirical way. At very beginning, when the network traffic was modeled by Poisson, the mean-variance relationship is very simple: $\sigma^2 = E[8]$. However, when W.E.Leland et al.[5] did the pioneering work on self-similarity, Poisson model failed[7], so did the relationship. Consequently, many papers have done many observations to find new form of the relationship. They all empirically agree that the relationship exists and its form is power law: $\sigma^2 = \varphi E^c[1]$, where φ and c are constants. However, c is assigned different values in different network environments. J.Cao et al.[1] choose 2 as approximate integer of c ; A.Medina et al. [6] conclude that c varies between 0.5 and 4; A.Gunna et al.[3] find that the values of c are 1.5 and 1.6 respectively on European and North American core-networks; I.Juva et al. [4] choose 1.5 as the best value of c . Thus, we also choose proper value of c by observing actual network traffics.

3 OD Count Estimation Algorithm

3.1 Mathematical Formulation

First of all, we provide mathematical descriptions of network topology and traffic stochastic processes to make the following discussion convenient. Consider a link layer topology that only consists of interconnected network devices, in which all the interfaces can be divided into two sets: one is for interconnection between devices and the other one is called network entries through which network traffics flow. For any entry x , let $Fi(x)$ denote the incoming link count stochastic process through x , and $Fo(x)$ correspond to the outgoing process. Since the OD count is discussed on link layer, the origin and destination nodes correspond to the network entries. Given any different entries x and y , let $F(x, y)$ denote OD count stochastic process which passes the network from x to y . Through periodical measuring by interval T seconds, we can get a sequence of link count samples for each entry whose element is defined as the entry's average speed. Thus the problem to be resolved by our algorithm can be described as follows: let N denote the length of the sample, given the samples of all network entries in time segment $[0, NT]$, for two designated interfaces x and y , estimate $E(F(x, y))$ and $E(F(y, x))$ in time segment $[0, NT]$.

3.2 Create Equivalent Network

We first create an equivalent network to make the following estimation simpler. There are two reasons: (1) exclude irrelevant network topology. Generally, we assume that the absolute majority of network traffic is unicast traffic and some faulty factors can be ignored such as the packet loss and so on. This implies that a network's total amount of incoming bytes should be equal to its total amount of outgoing bytes. However, the larger the scale of network is, the heavier the cumulation of those faulty factors is. Therefore, we should select a small

subnetwork which excludes irrelevant network topology to reduce this cumulative effect. (2) Minimize computation. There are $n * (n - 1)$ OD counts in a n -Entry network, amount which only two OD counts are required. Thus we should minimize the value of n in order to reduce the computation. Given two entries x and y . We create equivalent network by two steps.

(1) Choose one subnetwork N_0 which includes x and y as its entries and at least includes all the network devices and links that the traffics between x and y pass through. Define set I include all N_0 's entries except x and y . As shown in Table 1 we can find that in N_0 , there are always 7 sorts of OD counts.

Table 1. 7 Sorts of OD Counts

No.	Incoming	Outgoing
1	x	y
2	x	$\forall z, z \in I$
3	y	x
4	y	$\forall z, z \in I$
5	$\forall z, z \in I$	x
6	$\forall z, z \in I$	y
7	$\forall z, z \in I$	$\forall z, z \in I$

(2) Create equivalent three-entry network. We create a virtual network including only 3 entries: x , y and w . w 's incoming and outgoing link count samples are defined by $Fi(w) = \sum_{i \in I} Fi(i)$ and $Fo(w) = \sum_{i \in I} Fo(i)$.

Define S as a vector of link count stochastic processes and F as a vector of OD count stochastic processes. We have:

$$S = AF, \quad (1)$$

where

$$S = \begin{bmatrix} Fi(x) \\ Fi(y) \\ Fi(w) \\ Fo(x) \\ Fo(y) \\ Fo(w) \end{bmatrix}, A = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}, F = \begin{bmatrix} F(x, y) \\ F(x, w) \\ F(y, x) \\ F(y, w) \\ F(w, x) \\ F(w, y) \\ F(w, w) \end{bmatrix}$$

We can find that there are only 7 unknown OD counts passing through the virtual network. Additionally, define λ as mean of F and Y as mean of S , we also have:

$$Y = A\lambda, \quad (2)$$

where

$$\begin{aligned} Y = E(S) &= [E(Fi(x)) \ E(Fi(y)) \ E(Fi(w)) \ E(Fo(x)) \ E(Fo(y)) \ E(Fo(w))]' \\ \lambda = E(F) &= [E(F(x, y)) \ E(F(x, w)) \ E(F(y, x)) \ E(F(y, w)) \\ &\quad E(F(w, x)) \ E(F(w, y)) \ E(F(w, w))]' \end{aligned}$$

Therefore, we can achieve the goal by resolving the value of λ_1 and λ_3 .

3.3 OD Count Estimation

Three-entry network owns 6 link count processes, which implies 21 link count covariances (including 6 link count variances). And 7 OD count processes correspond to 28 OD count covariances (including 7 OD count variances). Let Z denote vector of link count covariances and X denote vector of OD count covariances. According to matrix A , we create the linear relationship between Z and X now.

Choose row vectors A_i and A_j , where $i \leq j$. We can create 7×7 Matrix $B(i, j)$ whose elements are defined by:

$$B(i, j)_{m,n} = \begin{cases} A_{i,m} \times A_{j,n} + A_{i,n} \times A_{j,m} & (m < n) \\ A_{i,m} \times A_{j,n} & (m = n) \\ 0 & (m > n) \end{cases}$$

$B(i, j)$ is obviously an upper triangular matrix. Create 1×28 row vector $C(i, j)$ by rearranging the upper triangular elements of $B(i, j)$.

$$C(i, j) = [B(i, j)_{1,1} \ B(i, j)_{1,2} \ \cdots \ B(i, j)_{1,7} \\ B(i, j)_{2,2} \ \cdots \ B(i, j)_{2,7} \ B(i, j)_{3,3} \ \cdots \ B(i, j)_{7,7}]$$

Then we get the relation between Z and X :

$$Z = DX, \quad (3)$$

where

$$\begin{aligned} Z &= [Cov(y_1, y_1) \ \cdots \ Cov(y_1, y_6) \ Cov(y_2, y_2) \ \cdots \ Cov(y_6, y_6)]' \\ D &= [C(1, 1) \ \cdots \ C(1, 6) \ C(2, 2) \ \cdots \ C(6, 6)]' \\ X &= [Cov(F_1, F_1) \ \cdots \ Cov(F_1, F_7) \ Cov(F_2, F_2) \ \cdots \ Cov(F_7, F_7)]' \end{aligned}$$

$Cov(F_k, F_k) = \sigma_k^2 (k = 1, 2, \dots, 7)$ should be resolved. We prefer multivariate linear regression to estimate them. Because equation (3) is still an indeterminate equation. Therefore, another assumption is required. Generally, OD count processes are assumed as independent between each other. According to this assumption, all OD count covariances should be 0. Thus, we can get a simple form of equation (3).

$$Z = RX', \quad (4)$$

where

$$R = R_0 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}, X' = X'_0 = \begin{bmatrix} Cov(F_1, F_1) \\ Cov(F_2, F_2) \\ Cov(F_3, F_3) \\ Cov(F_4, F_4) \\ Cov(F_5, F_5) \\ Cov(F_6, F_6) \\ Cov(F_7, F_7) \end{bmatrix}$$

Then the result of X'_0 can be estimated by multivariate linear regression: $X' = (R^T R)^{-1} R^T Z$. Because a variance should be nonnegative, all the elements of X'_0 should be nonnegative. However, multivariate linear regression method can not satisfy this restriction. Based on the experiments on actual network, we often get negative results. In our opinion, the main reason is: It is too strong to assume that all the actual OD count processes are independent between each other. For example, TCP communication requires the receiver to send ACK packets to guarantee reliability. In this case two opposite OD count processes may be strongly correlated. In addition, because of network config, the trace of some network traffics may pass through different OD pairs whose counts may be strongly correlated as well[9].

To resolve the upper problem, we assume that 21 OD count covariances consists of k nonzero values and $21 - k$ zero values. Thus there are C_{21}^k possible combinations of k covariances, which implies that many nonzero covariances cost a great deal of computation. Contrarily, a few nonzero covariances can not achieve acceptable results. Therefore, we empirically restrict $3 \leq k \leq 7$. Before proposing the complete estimation algorithm, we build column vector X'_C and matrix R_C first. Define set V include the column numbers of $Cov(F_i, F_i)$, where $1 \leq i \leq 7$. Then create X'_C by up to down selecting elements from X whose row number is not in V and create R_C by left to right selecting columns from D whose column number is not in V . Then D and X can be expressed as:

$$D = [R_0 \ R_C], X = \begin{bmatrix} X'_0 \\ X'_C \end{bmatrix}$$

Given Z , the algorithm to estimate OD count variances is defined as follows:

```

Function resolve( $Z$ )
begin
   $result := [0, 0, 0, 0, 0, 0, 0]'$ ;
   $Q := \infty$ ;
  for  $i := 3$  to 7 do
    for  $(j_1, j_2, \dots, j_i)$  as combination of  $1, 2, \dots, 21$ 
      Build  $R_T$  by selecting columns  $j_1, j_2, \dots, j_i$  from  $R_C$ ;
      Build  $X'_T$  by selecting rows  $j_1, j_2, \dots, j_i$  from  $X'_C$ ;
      Set  $R$  as  $[R_0 \ R_T]$ ;
      Set  $X'$  as  $\begin{bmatrix} X'_0 \\ X'_T \end{bmatrix}$ ;
       $X' := (R^T R)^{-1} R^T Z$ ;
      if not all elements of  $X'_0$  are nonnegative then
        continue;
      end if
       $q :=$  the residual sum of squares of  $Z = R X'$ ;
      if  $Q > q$  then
         $Q := q$ ;
         $result := X'_0$ ;
      end if
    end for
  end for
  if  $result$  is not  $[0, 0, 0, 0, 0, 0, 0]'$  then
    break;
  end if
end for
return  $result$ ;
end

```

After achieving X'_0 , the OD count mean values can be obtained directly from the mean-variance relationship. However, equation (2) is a stronger condition. Because $rank(A) = 5$, equation (2) is a redundant equation. λ_i can be linearly expressed by λ_1 and λ_3 by removing the last row of matrix A . Then we can get equation:

$$G = HL \quad (5)$$

where

$$G = \begin{bmatrix} \varphi^{-\frac{1}{c}} X'_1{}^{\frac{1}{c}} \\ \vdots \\ \varphi^{-\frac{1}{c}} X'_7{}^{\frac{1}{c}} \end{bmatrix}, H = [\alpha \ \beta \ \gamma] = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 0 & Y_1 \\ 0 & 1 & 0 \\ 0 & -1 & Y_2 \\ 0 & -1 & Y_4 \\ -1 & 0 & Y_5 \\ 1 & 1 & Y_3 - Y_4 - Y_5 \end{bmatrix}, L = \begin{bmatrix} \lambda_1 \\ \lambda_3 \\ 1 \end{bmatrix}$$

Because φ is still unknown, we use multivariate linear regression again to avoid resolving $\varphi^{-(1/c)}$. Select row H_i and row H_j of equation (5), where $i \neq j$,

we have $G_i/G_j = (\alpha_i\lambda_1 + \beta_i\lambda_3 + \gamma_i)/(\alpha_j\lambda_1 + \beta_j\lambda_3 + \gamma_j)$, i.e. $X_i'^{\frac{1}{c}}\gamma_j - X_j'^{\frac{1}{c}}\gamma_i = (X_j'^{\frac{1}{c}}\alpha_i - X_i'^{\frac{1}{c}}\alpha_j)\lambda_1 + (X_j'^{\frac{1}{c}}\beta_i - X_i'^{\frac{1}{c}}\beta_j)\lambda_3$. Therefore, the equation can be denoted by:

$$W = U\Lambda \quad (6)$$

where

$$W = \begin{bmatrix} X_1'^{\frac{1}{c}}\gamma_2 - X_2'^{\frac{1}{c}}\gamma_1 \\ X_1'^{\frac{1}{c}}\gamma_3 - X_3'^{\frac{1}{c}}\gamma_1 \\ \dots \\ X_1'^{\frac{1}{c}}\gamma_7 - X_7'^{\frac{1}{c}}\gamma_1 \\ X_2'^{\frac{1}{c}}\gamma_3 - X_3'^{\frac{1}{c}}\gamma_2 \\ \dots \\ X_6'^{\frac{1}{c}}\gamma_7 - X_7'^{\frac{1}{c}}\gamma_6 \end{bmatrix}, U = \begin{bmatrix} X_2'^{\frac{1}{c}}\alpha_1 - X_1'^{\frac{1}{c}}\alpha_2 & X_2'^{\frac{1}{c}}\beta_1 - X_1'^{\frac{1}{c}}\beta_2 \\ X_3'^{\frac{1}{c}}\alpha_1 - X_1'^{\frac{1}{c}}\alpha_3 & X_3'^{\frac{1}{c}}\beta_1 - X_1'^{\frac{1}{c}}\beta_3 \\ \dots & \dots \\ X_7'^{\frac{1}{c}}\alpha_1 - X_1'^{\frac{1}{c}}\alpha_7 & X_7'^{\frac{1}{c}}\beta_1 - X_1'^{\frac{1}{c}}\beta_7 \\ X_3'^{\frac{1}{c}}\alpha_2 - X_2'^{\frac{1}{c}}\alpha_3 & X_3'^{\frac{1}{c}}\beta_2 - X_2'^{\frac{1}{c}}\beta_3 \\ \dots & \dots \\ X_7'^{\frac{1}{c}}\alpha_6 - X_6'^{\frac{1}{c}}\alpha_7 & X_7'^{\frac{1}{c}}\beta_6 - X_6'^{\frac{1}{c}}\beta_7 \end{bmatrix}, \Lambda = \begin{bmatrix} \lambda_1 \\ \lambda_3 \end{bmatrix}$$

Equation (6) is obviously a redundant equation. Thus based on multivariate linear regression, we can estimate Λ by:

$$\Lambda = (U^T U)^{-1} U^T W. \quad (7)$$

4 Verification

In this section, we apply our algorithm in the official network of our laboratory, whose topology is shown in Fig.1. By observing actual network traffics, we first assign a proper value of c and then verify our algorithm by comparing the estimation result with the OD count data measured by NetFlow.

We periodically collect link byte count samples by interval $T = 30s$, and evaluate the mean and variance of the samples with $N = 10$. Five typical log-log scale mean-variance relationship graphs are drawn together in Fig.1. By least squares approximation we set $c = 1.8$, which is a proper value for our laboratory network. Thus, the mean-variance relationship is $\sigma^2 = \varphi E^{1.8}$.

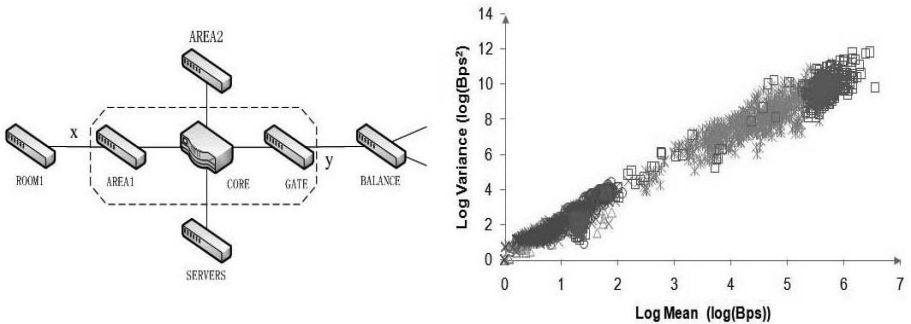


Fig. 1. Network topology and Log-log relationship

Consequently, we apply our algorithm to estimate the OD counts between ROOM1 and BALANCE. The designated entries and subnetwork are shown in Fig.1. The subnetwork has 18 entries in total, which corresponds to $18 \times 17 = 306$ OD pairs. According to the method of creating three-entry network, the mean of link counts are shown in Fig.2. By applying the algorithm, the estimation results and the Netflow measurement data are compared in Fig.3. And we verify that the estimation results are basically identical with the exact data.

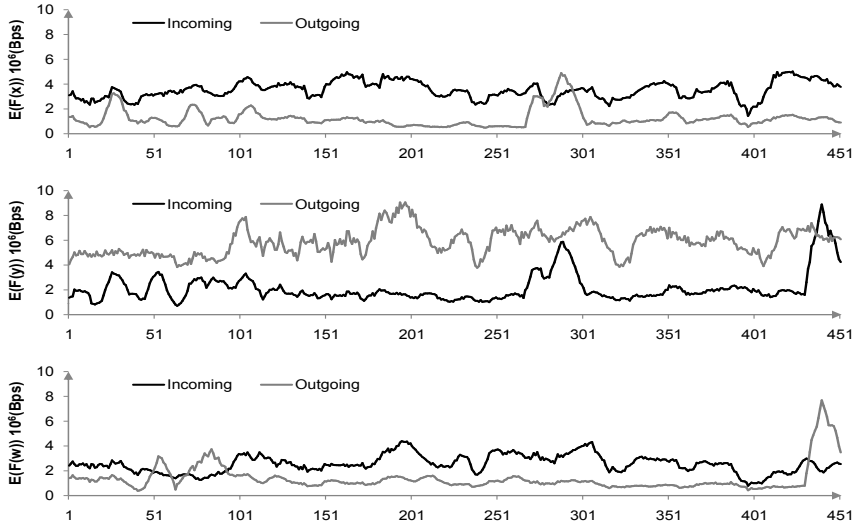


Fig. 2. Mean link counts of three-entry Network

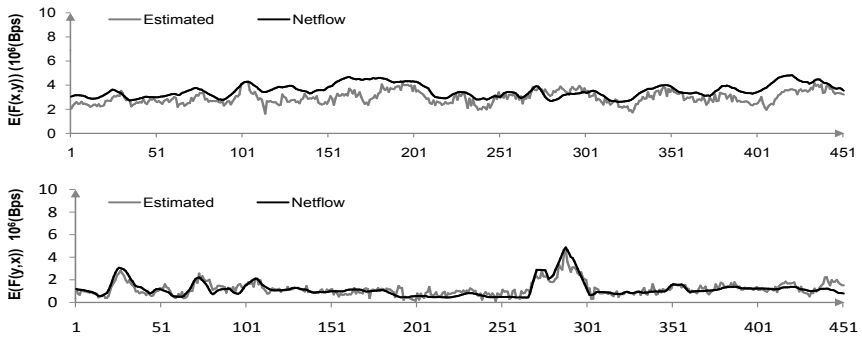


Fig. 3. Verify estimated values by NetFlow

5 Conclusion

In this paper, we provide a new estimation algorithm to estimate designated OD counts on IP network link layer. Compared with the existing estimation algorithms, our algorithm is more suitable for the networks having many entries. The algorithm is designed based on link count data which is easy to collect. This property makes our algorithm easy to apply in reality. The algorithm includes two steps: (1) build three-entry network to simplify the computation. (2) Estimate OD counts by multivariate linear regression. Then, we apply our algorithm in our laboratory official network, and verify it by comparing with the OD count data measured by NetFlow.

As for the future work, we plan to derive the theoretical mean-variance relationship from self-similar traffic model. This will theoretically support all the algorithms that use this relationship as the basic estimation assumption.

References

1. Cao, J., Davis, D., Wiel, S., Yu, B.: Time-varying network tomography: Router link data. *Journal of the American Statistical Association* 95(452) (December 2000)
2. Cisco Systems. Introduction to Cisco IOS NetFlow - A Technical Overview
3. Gunnar, A., Johansson, M., Telkamp, T.: Traffic matrix estimation on a large ip backbone - a comparison on real data. In: *Proceeding ACM IMC 2004*, October 2004, pp. 149–160. ACM Press, New York (2004)
4. Juva, I., Vaton, S., Virtamo, J.: Quick traffic matrix estimation based on link count covariances. In: *IEEE International Conference on Communications (ICC 2006)*, vol. 2 (June 2006)
5. Leland, W.E., Taqqu, M.S., Willinger, W., Wilson, D.V.: On the self-similar nature of ethernet traffic (extended version). *IEEE ACM Transactions on Networking (TON)* 2(1) (February 1994)
6. Medina, A., Taft, N., Salamatian, K., Bhattacharyya, S., Diot, C.: Traffic matrix estimation: existing techniques and new directions. *ACM SIGCOMM Computer Communication Review* 32(4) (October 2002)
7. Paxson, V., Floyd, S.: Wide-area traffic: The failure of poisson modeling. *IEEE/SlashACM Transactions on Networking* 3(3), 226–244 (1995)
8. Vardi, Y.: Network tomography: estimating source-destination traffic intensities from link data. *Journal of the American Statistical Association* 91(433) (March 1996)
9. Errmailli, V., Crovella, M., Taft, N.: An Independent-Connection Model for Traffic Matrices. In: *Proceeding ACM IMC 2006*, October 2006. ACM Press, New York (2006)
10. Claise, B. (ed.): Cisco Systems NetFlow Services Export Version 9, RFC3954 (October 2004), <http://www.ietf.org/rfc/rfc3954.txt>
11. Phaál, P.: InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks, RFC3176 (September 2001), <http://www.ietf.org/rfc/rfc3176.txt>

An Efficient Approach for Analyzing Multidimensional Network Traffic

Jia Yang, Hao Ma, Bei Zhang, and Ping Chen

Peking University, Beijing 100871, P.R. China
{yangj,mah,zhp,pchen}@pku.edu.cn

Abstract. Identifying dominant network flows is important for network anomaly detection. Estan et al. proposed an algorithm that effectively detects dominant network flows by constructing multidimensional clusters based on a “natural hierarchy” existing in the five-tuple information of network flows. Wang et al. improved this algorithm by significantly reducing its computational complexity. In practice, however, the algorithm’s execution time may be relatively long when handling large volumes of traffic with a low threshold. In this paper, we introduce a practical technique that further improves the time efficiency of Wang et al.’s algorithm. Our approach simplifies network traffic’s hierarchical structure by utilizing local IP subnet information. The comparative performance of our approach and Wang et al.’s algorithm is evaluated using real NetFlow data collected at a large campus network. The experimental results demonstrate that our algorithm is much more time efficient than Wang et al.’s algorithm.

1 Introduction

Network anomaly detection is a very important topic for network management. It thus has been studied extensively in recent years. However, identifying network anomaly efficiently and accurately still remains a challenging task, considering the ever-increasing volumes of network traffic and complexity of usage models.

Generally, network managers need to construct a “normal” network traffic model based on network traffic usages in order to detect anomalies. As discussed in [3], a often-used method is to classify network traffic by application types, such as HTTP, FTP, SMTP, P2P protocols, etc., study the usage for each type of traffic over a specified time interval, and then infer a set of standard patterns that indicate normal network usage. Compared with the standard patterns, network anomalies may be identified.

However, it is increasingly difficult to accurately classify network traffic by application types. The traditional and most time efficient port-based (combined with protocol type information that can be extracted from IP header fields) approaches often fail to recognize traffic’s application types, as new applications may use ports not included in the network managers’ predefined port sets, and more and more applications are using dynamically assigned or customizable ports (e.g., the popular p2p programs PPLive [6] and eMule [5]).

In order to classify traffic more accurately, *Deep Packet Inspection (DPI)* [8] can be used to examine not only the header part, but also the data part of packets. However, as a computationally expensive technique, generally DPI is not suitable to be deployed in-line to handle large volumes of traffic. Another problem with DPI is that it can not inspect encrypted flows. Unfortunately, many applications deliberately encrypted their traffic for security reasons or to avoid surveillance (again, use eMule as an example).

Network anomalies may also be detected by using flow mining techniques [3,4,1,2]. As discussed in [4], some approaches mine traffic in order to identify significant flows [3,4,2], while some aim to find *deviant* flows [1]. There are different methods to accomplish the same goal. For example, significant flows can be identified by approximate counting [2], or by exact counting [3,4].

Identifying dominant network flows is a practical method for network anomaly detection. This paper will focus on the techniques proposed in [3,4], as they are effective to handle large volumes of traffic without introducing errors (compared with methods based on approximate counting). In [3], Estan et al. proposed an algorithm (i.e., *Autofocus*) that detects dominant network flows (the significance of flows is determined by a customizable threshold) by constructing multidimensional traffic clusters. It can capture hierarchical structure of significant traffic over a single dimension (e.g., source or destination IP), or over multi-dimensions (e.g., the *five-tuple*: source IP, destination IP, source port, destination port, and protocol).

Later, Wang et al. significantly reduced the computational complexity of Estan et al.'s algorithm [4], which makes it possible for this algorithm to be used in real-time scenarios. In this paper, we introduce a method that further improves Wang et al.'s algorithm. Our approach exploits local IP subnet information to simplify trees and graphs that represent the unidimensional and multidimensional characteristics of traffic. We are not aware of any other work that has proposed this technique.

In this paper, we first give a brief introduction to the algorithms discussed in [3,4], and then present our approach, followed by experiments that evaluate our method using real NetFlow [7] data collected at a large campus network.

2 Multidimensional Traffic Clustering

Generally speaking, most network anomalies change network traffic over a number of measurement dimensions. Valuable information may thus be lost if traffic is only characterized along one single dimension. However, measurement methods based on multidimensional traffic clusters are inherently computationally expensive. In order to efficiently build the multidimensional traffic clusters, a multidimensional traffic clustering algorithm was introduced by Estan et al. in [3]. The algorithm can be tuned to focus on significant flows while ignore unimportant ones. This strategy is consistent with the fact that the best practice for network managers to monitor their networks is to focus on significant flows. There is too much information carried by network traffic, it is thus practically impossible for network managers to handle excessively detailed information.

2.1 Estan et al's Algorithm

In [3], the five-tuple (source IP, destination IP, source port, destination port, and protocol) is used to define traffic clusters. For the five fields, a simple method was proposed to utilize their “natural hierarchies”. IP addresses’ hierarchy is defined by (*) that represents all IP addresses, and prefixes with different lengths (from 8 to 32). An IP prefix is denoted in the form of $x.x.x.x/n$ (using the *CIDR* notation [9]), where $x.x.x.x$ is the prefix bits and trailing bits (trailing bits are set to 0), and n is the number of bits in the prefix. Port numbers’ hierarchy is defined by (*) that represents all port numbers, their range, i.e., whether they are high ports (port number ≥ 1024) or low ports (port number < 1024), and their exact values. Protocols’ hierarchy is defined by (*) that includes all types of protocols, and the exact protocols (e.g., UDP and TCP).

Based on the hierarchies discussed above, a graph (more precisely, it is a *Directed Acyclic Graph* [11] with a single root node) can be constructed to reveal the multidimensional characteristics of network traffic. In the graph, there is a root node (SrcIP=*, DstIP=*, SrcPort=*, DstPort=*, Proto=*) that represents all traffic. The root node has a number of child nodes, such as (SrcIP=*, DstIP=*, SrcPort=*, DstPort=*, Proto=TCP), which represents all TCP traffic, and (SrcIP=*, DstIP=*, SrcPort=H, DstPort=*, Proto=*), which represents all traffic coming from high ports.

It is difficult to show the full view of a graph in a figure. Thus we only depict a small part of a graph based on two dimensions (source IP addresses and source ports) in Figure 1.

In Figure 1, node (0.0.0.0/1, *) represents all traffic coming from IP prefix 0.0.0.0/1. Note that, we usually use prefix lengths from 8 to 32 in practice. The reason that we use prefix lengths 1 and 2 in the figure is to simplify the graph.

As demonstrated in Figure 1, the structure of the graph is much more complicated than that of a unidimensional tree:

1. One node may have multiple parent nodes (e.g., node 6 has two parent nodes, node 2 and node 4).
2. Nodes at the same level of the graph may have overlapped flow sets (e.g., node 6 and node 9 have overlapped flow sets. Note that these two nodes are not only at the same level, but also share a common parent node).

Estan et al. introduced methods for constructing unidimensional clusters as well as multidimensional clusters [3]. In the case of unidimensional clustering, it is easy to construct unidimensional trees for ports or protocols, as the number of possible values of ports or protocols is relatively small. We can set a counter for each of the possible values in memory, such that the counter can be accessed and updated efficiently. The number of possible values for IP addresses is too large. It is thus impractical to assign each value a counter in memory. In [3], a bottom-up approach was proposed to build unidimensional trees for IP addresses, i.e., a tree is built from its leaf nodes to root node. Significant nodes are determined based on a predefined threshold (e.g., %5).

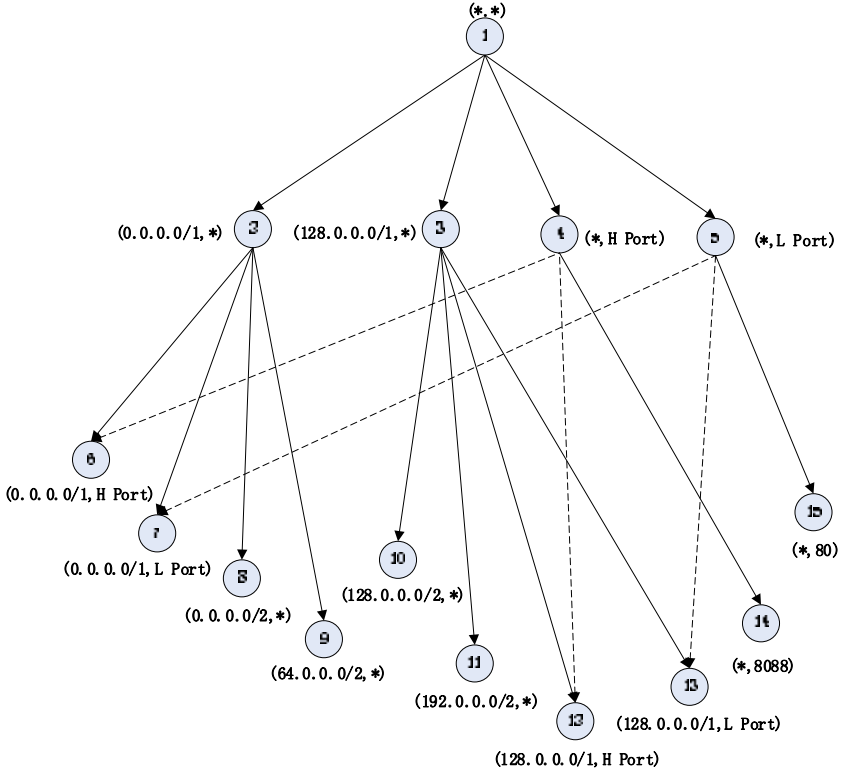


Fig. 1. Partial view of a two-dimensional graph over source IP addresses and source ports. Each node in the graph is denoted by (source IP, source port). Note that “H port” represents high ports, and “L port” represents low ports.

It is much more computationally expensive to build a graph than to build a tree. In [3], a graph only consists of significant nodes (note that the root node is always regarded as a significant node), and is constructed from root to leaves. Several techniques were introduced in order to reduce computational complexity (see [3] for more information), such as using unidimensional trees to help identify the significant nodes in a graph (if a unidimensional ancestor of a node is insignificant, then the node can not be significant), and determining the significance of nodes by checking their parent nodes (all ancestors of a significant node should also be significant).

2.2 Wang et al’s Algorithm

Wang et al. improved Estan et al.’s algorithm by significantly reducing its computational complexity [4]. The main modifications to the original algorithm include:

1. Top-down unidimensional clustering. Unidimensional trees are constructed from root to leaves. However, in [3] unidimensional trees for IP addresses are built up from leaves (bottom-up).

2. Flow subset paradigm for multidimensional clustering. Assume that A and B are two nodes in a multidimensional graph, and A is one of B 's parents. Let F_A denote the set that consists of all flows belonging to A , and F_B all the flows belonging to B . In order to identify F_B , we only need to search in F_A instead of all flows, as $F_B \subseteq F_A$.
3. Minimal parent strategy for multidimensional clustering. Assume that in a multidimensional graph, node B has n ($n > 1$) parent nodes, $A_1, A_2, \dots, A_n$. Let $|F_{A_i}|$ denote the number of flows that belong to A_i , where $1 \leq i \leq n$. As discussed before, F_B can be determined by searching in any of its parent node, F_{A_i} . In order to reduce computational complexity, we should search in F_{A_j} ($1 \leq j \leq n$), such that $|F_{A_j}| \leq |F_{A_i}|$ for $1 \leq i \leq n$.
4. Complement strategy for multidimensional clustering. Let node A have multiple child nodes, which can be put into a number of *distinct sibling groups*. In each group, the nodes' flow sets are "mutually exclusive and collectively exhaustive" [4] of the parent node's flow set. For example, if $D = \{B_1, B_2, \dots, B_m\}$ ($m > 1$) is a distinct sibling group of A , then (1) $F_{B_j} \cap F_{B_k} = \emptyset$, where $j \neq k$ and $B_j, B_k \in D$, and (2) $F_{B_1} \cup F_{B_2} \cup \dots \cup F_{B_m} = F_A$. Assume that F_{B_1} has been identified by searching in F_A . In order to identify F_{B_2} , we only need to search in $(F_A - F_{B_1})$. If $F_{B_1}, F_{B_2}, \dots, F_{B_i}$ ($i < m$) have been identified in F_A , we only need to search in $(F_A - F_{B_1} - F_{B_2} - \dots - F_{B_i})$ in order to get $F_{B_{i+1}}$.

Experimental results given in [4] show that the improved algorithm is much more time efficient than the original one.

3 Improving the Multidimensional Traffic Clustering Algorithm

In this section, we introduce the improvements to Wang et al.'s algorithm, which makes unidimensional and multidimensional traffic clustering even faster.

We implemented the algorithm described in [4] from scratch and used it to monitor the traffic data collected at the main campus of Peking University. The algorithm demonstrated its power for efficiently capturing significant flows and providing valuable multidimensional information that can help network managers quickly identify network anomalies. However, we noticed that, the algorithm's execution time may be relatively long when handling large volumes of traffic with a low threshold. Setting a high threshold can eliminate this problem, because the higher the threshold we use, the more "insignificant" flows are ignored. But if the threshold is too high, we are facing the risks of ignoring large flows which are not "significant" enough to be identified.

Therefore, it is necessary to further improve the time efficiency of the algorithm. We accomplished this goal by exploiting local IP subnet information.

3.1 Algorithm Description

Network managers monitor traffic between different networks, e.g., traffic between a local network and the Internet, or traffic between two local networks.

Local IP subnet information, which is well known to network managers, can be utilized to improve the algorithm's time efficiency.

Assume that we are monitoring traffic between a local network and the Internet. The local network consists of three subnet, 192.168.100/24, 192.168.200/24, and 10.100.252.0/22. Based on this assumption, the major modifications to the algorithm in [4] are introduced below:

1. In the case of unidimensional traffic clustering, two sets of unidimensional trees are constructed. One set of trees represent the unidimensional clustering for inbound traffic, and the other set of trees represent that for outbound traffic.
 - (a) When constructing the unidimensional tree for inbound traffic over destination IP addresses, let the child nodes of the root node be 192.168.100/24, 192.168.200/24, and 10.100.252.0/22. In the original algorithm, however, there are 256 child nodes of the root node, i.e., nodes $\{i.0.0.0/8\}$ for $0 \leq i \leq 255$. Starting from the 256 nodes, the original algorithm will eliminate the insignificant nodes, generate the respective child nodes at different levels, and eventually get to nodes 192.168.100/24, 192.168.200/24, and 10.100.252.0/22, if these are significant nodes. Note that at least one of the three node is significant, unless the threshold value is unreasonably large (i.e., larger than 33.3%). Hence, utilizing local IP subnet information enables us to simplify the tree structure and construct the tree more efficient.
 - (b) Similarly, when constructing the unidimensional tree for outbound traffic over source IP addresses, let the child nodes of the root node be 192.168.100/24, 192.168.200/24, and 10.100.252.0/22. It thus speeds up the process of tree generation, following the same arguments discussed in Step 1a.
2. In the case of multidimensional traffic clustering, two graphs are constructed, one graph (G_{in}) representing the multidimensional clustering for inbound traffic and the other one (G_{out}) representing that for outbound traffic.
 - (a) In the process of building G_{in} for inbound traffic, considering the destination IP dimension only, let the child nodes of the root node be 192.168.100/24, 192.168.200/24, and 10.100.252.0/22. Following the same arguments discussed in Step 1a, this strategy simplifies the graph structure and speeds up the construction process. As discussed before, the structure of a graph is much more complicated than that of a tree. Therefore, simplifying the graph structure may significantly improve the time efficiency of multidimensional clustering.
 - (b) When constructing G_{out} for outbound traffic, and considering the source IP dimension only, let the child nodes of the root node be 192.168.100/24, 192.168.200/24, and 10.100.252.0/22. This technique may greatly improve the time efficiency of multidimensional clustering, following the same arguments discussed before.

The improvements described above are based on the assumption that the traffic is between a local network and the Internet. If the traffic is between two local

networks, the structures of unidimensional trees and multidimensional graphs can be further simplified, as both the source IP addresses and the destination IP addresses are within a small range.

3.2 Experiments

This section discusses the comparative performance of the time efficiency of Wang et al.'s algorithm and ours. We first implemented Wang et al.'s algorithm as a C program, called *cluster1* in order to facilitate the discussion. Next, based on *cluster1* we implemented the improvements discussed in Section 3.1. The new program is called *cluster2*.

The comparison was performed on a *Red Hat* PC (Red Hat Enterprise Linux Server release 5) with 4GB memory and an Intel Xeon CPU (3.00GHz).

In the experiments, we used all the NetFlow files collected at the main campus of Peking University on May 11th 2008. The total number of the NetFlow files is 288, each file recording the inbound and outbound flows during a five-minute time interval.

The main campus of Peking University has one of the largest campus networks in China. The average numbers of flows, packets, and bytes in each of the 288 NetFlow files generated on May 11th 2008 are 3.16×10^6 , 150.55×10^6 , and 100.36×10^9 , respectively. The original NetFlow files were compressed binary files. Before the experiments, all the files were uncompressed, but remained in binary format. The total size of all these uncompressed files is 58.26×10^9 bytes.

We used *cluster1* and *cluster2* (utilizing the IP subnet information of the campus network) to construct unidimensional and multidimensional traffic clusters for each of the 288 NetFlow files, respectively. Note that in the experiments, the traffic counting is based on the number of flows.

Table 1 gives the average execution times with the respective standard deviations for the two programs to compute unidimensional clusters. As shown in this table, *cluster2* runs much faster than *cluster1*, which is consistent with the discussion in Section 3.1.

Table 2 shows the average execution times with the respective standard deviations for the two programs to analyze multidimensional traffic. As demonstrated in this table, *cluster2* runs much faster than *cluster1*. Note that the improvement of *cluster2* over *cluster1* is even more significant than that in the case of unidimensional clustering (shown in Table 1).

Both tables demonstrate that the execution times for *cluster2* and *cluster2* decrease with increasing threshold values. The reason is that, a high threshold can simplify the structure of a tree or graph (as there are only a small number of significant nodes in the tree or graph).

In the experiments, we used seven thresholds, %1, %2, %5, %7, %10, %15, and %20. In practice, however, we would suggest to use thresholds between %1 and %5 in order to produce meaningful results for network managers.

Figure 2 gives a snapshot of the partial result of the multidimensional outbound traffic clustering for a NetFlow file (using threshold 1%). Let $|flows_{all}|$ denote the number of all flows, $|packets_{all}|$ the number of all packets, and

Table 1. Averaged execution times (in seconds) with the respective standard deviation values for *cluster1* and *cluster2* to construct unidimensional traffic clusters for each of the 288 NetFlow files at different thresholds

threshold		1%	2%	5%	7%	10%	15%	20%
Averaged Execution Time	<i>cluster1</i>	11.96	10.61	8.76	8.19	7.71	7.50	6.98
	<i>cluster2</i>	5.21	3.91	2.68	2.32	2.15	2.08	2.07
Standard Deviation	<i>cluster1</i>	1.96	1.75	1.42	1.34	1.25	1.23	1.17
	<i>cluster2</i>	0.88	0.67	0.44	0.38	0.35	0.33	0.33

Table 2. Averaged execution times (in seconds) with the respective standard deviation values for *cluster1* and *cluster2* to construct multidimensional traffic clusters for each of the 288 NetFlow files at different thresholds

threshold		1%	2%	5%	7%	10%	15%	20%
Averaged Execution Time	<i>cluster1</i>	156.98	93.40	43.30	35.92	29.96	22.95	20.16
	<i>cluster2</i>	25.88	15.48	6.89	5.22	4.40	3.65	3.43
Standard Deviation	<i>cluster1</i>	28.65	16.55	7.14	6.02	5.22	3.52	3.20
	<i>cluster2</i>	4.58	2.86	1.17	0.88	0.76	0.55	0.53

$|bytes_{all}|$ the number of all bytes. For a node, let $|flows_{node}|$ denote its flow count, $|packets_{node}|$ its packet count, and $|bytes_{node}|$ its bytes count. The fields in each line are the respective node (source IP, destination IP, source port, destination port, protocol), $\frac{|flows_{node}|}{|flows_{all}|} \times 100\%$, $\frac{|packets_{node}|}{|packets_{all}|} \times 100\%$, $\frac{|bytes_{node}|}{|bytes_{all}|} \times 100\%$, $|flows_{node}|$, $\frac{|packets_{node}|}{10^6}$ M, and $\frac{|bytes_{node}|}{10^6}$ M. Note that “M” represents 10^6 here.

As shown in Figure 2, there were significant flows that may indicate problems. For example, it seems that the host with IP address 162.105.146.12 was generating a large number of TCP flows (which contribute more than one percent of the total flows) to scan a large IP address range with destination port 80.

(162.105.83.0/25, *, H Port, H Port, *)	1.12%	0.68%	0.68%	14207	0.39M	301.06M
(162.105.99.9/32, *, H Port, H Port, *)	1.32%	0.75%	0.19%	16799	0.43M	85.00M
(162.105.99.0/24, *, H Port, H Port, TCP)	1.07%	0.45%	0.25%	13547	0.26M	112.14M
(162.105.99.0/24, *, H Port, H Port, UDP)	1.12%	1.19%	0.65%	14197	0.69M	288.25M
(162.105.105.0/24, *, H Port, *, *)	1.05%	0.73%	0.54%	13344	0.42M	239.34M
(162.105.106.0/24, *, H Port, *, *)	1.29%	1.39%	1.14%	16389	0.81M	508.99M
(162.105.106.0/24, *, *, H Port, *)	1.16%	1.24%	0.81%	14695	0.72M	360.46M
(162.105.106.0/24, *, *, *, TCP)	1.16%	1.36%	1.06%	14712	0.79M	471.02M
(162.105.107.0/24, *, H Port, H Port, *)	1.04%	0.34%	0.24%	13161	0.19M	107.51M
(162.105.112.0/24, *, H Port, H Port, *)	1.04%	0.74%	0.71%	13246	0.43M	313.87M
(162.105.113.0/24, *, H Port, *, *)	1.00%	0.43%	0.47%	12703	0.25M	208.40M
(162.105.114.0/24, *, *, *, *)	1.00%	0.83%	0.57%	12723	0.48M	254.77M
(162.105.129.16/28, *, H Port, L Port, *)	1.25%	0.10%	0.03%	15893	0.06M	12.13M
(162.105.129.0/26, *, *, *, TCP)	1.48%	1.54%	2.82%	18728	0.90M	1255.87M
(162.105.146.12/32, *, H Port, 80, TCP)	1.20%	0.25%	0.02%	15200	0.15M	9.22M
(162.105.146.24/32, *, 30610, H Port, UDP)	1.24%	0.21%	0.01%	15703	0.12M	5.81M
(162.105.146.16/28, *, H Port, *, TCP)	1.06%	0.23%	0.04%	13443	0.13M	19.37M
(162.105.146.0/26, *, *, H Port, TCP)	1.26%	0.12%	0.04%	15953	0.07M	18.82M

Fig. 2. Snapshot of the (partial) result of the multidimensional outbound traffic clustering for a NetFlow file (using threshold 1%)

4 Conclusions

In this paper, we briefly introduced a multidimensional traffic clustering algorithm proposed by Estan et al. [3] that utilizes the natural hierarchy existing in the five-tuple fields, and the improvements suggested by Wang et al. [4] which significantly improve the algorithm's time efficiency. Based on the work discussed in [4], we further improve the algorithm's time efficiency by exploiting local IP subnet information to simplify unidimensional tree structures as well as multidimensional graph structures.

In order to discuss the comparative performance of Wang et al.'s algorithm and ours, we implemented these two algorithms and use them to process real NetFlow data collected at the main campus of Peking University. The experimental results demonstrate that our approach is much more time efficient than Wang et al.'s algorithm for both unidimensional and multidimensional clustering.

References

1. Cormode, G., Muthukrishnan, S.: Whats new: Finding significant differences in network data streams. In: Proc. IEEE INFOCOM, pp. 1219–1232 (2004)
2. Manku, G., Motwani, R.: Approximate frequency counts over data streams. In: Proc. Int. Conf. Very Large Databases, pp. 346–357 (2002)
3. Estan, C., Savage, S., Varghese, G.: Automatically Inferring Patterns of Resource Consumption in Network Traffic. In: SIGCOMM 2003, pp. 137–148 (2003)
4. Wang, J., Miller, D.J., Kesidis, G.: Efficient Mining of the Multidimensional Traffic Cluster Hierarchy for Digesting, Visualization, and Anomaly Identification. In: IEEE JSAC, pp. 1929–1941 (2006)
5. The eMule project, <http://www.emule-project.net>
6. PPlive website, <http://www.pplive.com>
7. Cisco website, <http://www.cisco.com>
8. Definition of DPI from Wikipedia website, http://en.wikipedia.org/wiki/Deep_packet_inspection
9. Definition of CIDR from Wikipedia website, http://en.wikipedia.org/wiki/Classless_Inter-Domain_Routing
10. Introduction to the main campus network of Peking University (in Chinese), <http://www.pku.edu.cn/network/xyw/jj.htm>
11. Definition of Directed Acyclic Graph from MathWorld website, <http://mathworld.wolfram.com/AcyclicDigraph.html>

The Measurement of User's Subjective Assessment on Throughput Degradation in Bulk Data Transmission

Yasuhiro Sato¹, Shin'ichiro Kitano², Shingo Ata¹, and Ikuo Oka¹

¹ Graduate School of Engineering, Osaka City University
3-3-138 Sugimoto, Sumiyoshi-ku, Osaka 558-8585, Japan
{sato@n.,ata@,oka@}info.eng.osaka-cu.ac.jp

² TIS Inc.,
1-14-5 Kaigan, Minato-ku, Tokyo 105-8624, Japan
kitano@n.info.eng.osaka-cu.ac.jp

Abstract. Recent studies have focused on subjective assessment of a user, which assessment is referred as user-level QoS. The subjective assessment means a qualitative evaluation of a user. However, previous works have studied user's subjective assessment only in real-time traffic. Moreover, an absolute assessment has been performed in the subjective assessment of a user. Considering degradation of quality of network services, it is important how a user feels about degradation of the QoS in the usual network environment. In this paper, we measure user's subjective assessment in degradation of communication quality in bulk data transmission. To obtain the user's subjective assessment, we adopt *the constant method* and *the method of successive categories*. From results of the experiment, we clarify how much throughput degradation compared with the usual network environment makes a user be felt dissatisfaction with.

1 Introduction

Recently, real-time traffic, such as VoIP and video-streaming, and bulk traffic such as WWW and FTP coexist together due to the spread of the Internet, although they are quite different characteristics. This causes the degradation of QoS (Quality of Service), because all the packets are treated equally in IP network. To achieve the QoS of network services, Differentiated Service (Diff-Serv) [1] and Integrated Service (IntServ) [2] have been proposed as fundamental techniques. These techniques provide network-level QoS for ISP customers by making the resource reservation or differentiating the priority of forwarding packets. However, the network-level QoS guarantees metrics such as end-to-end delay, loss rate, and transfer bit rate etc., which metrics are from viewpoints of the network providers. Feedback from the customers in terms of quality of the network services is not considered in controlling the network-level QoS.

Therefore, recent studies have focused on subjective assessment of a user, which assessment is referred as user-level QoS. The subjective assessment means

a qualitative evaluation of a user such as “Conversing voices are broken up” and “A file downloading is extremely slow” etc. In [3], authors have proposed a scheme to assess the QoS in audio-video transmission. In addition, in [4], authors have shown how to control the performance of audio-video transmission by using the user-level QoS. In [5], authors have proposed the Dynamical User-Oriented QoS Framework based on dynamic QoS specification and a mapping scheme. This method maps the user-level QoS into the network-level QoS, and sets appropriate parameters of IntServ or DiffServ to achieve the user-level QoS. Authors in [6] have shown that an effective resource allocation for the customers can be achieved by using the user-level QoS.

As described above, previous works have studied user's subjective assessment only in real-time traffic. Moreover, in these literature, an absolute assessment has been performed in the subjective assessment of a user. That is, previous works have shown that appropriate absolute values of various metrics (typically, throughput of the service) for real-time applications obtained from the subjective assessment, and have controlled the network-level QoS based on the results of the literature. However, network environment of each user is greatly different depending on a user. In real network environment, we consider that the subjective assessment of a user is a comparative evaluation how much the QoS of network service degrades compared to the usual network environment of the user. Considering degradation of quality of network services, it is important how a user feels about degradation of the QoS in the usual network environment. When a competition of network resources occurs, the QoS of users are restricted, because available network resources are limited. In such situation, if we understand how much debasement of throughput of network services a user can perceive as degradation, more appropriate resource allocation for such users can be achieved. For this, it is necessary to quantitatively analyze the subjective assessment of users obtained by comparative evaluation.

In this paper, we measure user's subjective assessment in degradation of communication quality in bulk data transmission. We focus on throughput degradation of a network service which is easy for users to perceive as a network-level QoS. To obtain the user's subjective assessment, *the method of successive categories* [4] is a widely-used psychological method in previous works. Therefore, we adopt the method to measure user's subjective assessment. In the method of successive categories, users assess the QoS of network service by rating to several categories (typically, # of the categories is 5.). Moreover, to evaluate the QoS of network services more simply for users, we also consider a measurement by using *the constant method* [7], which is also a psychological method. In order to get users to make comparative evaluation, we have users be accustomed to a particular network environment which is assumed as the usual network environment of each user. In our experiment, we examine the effect of performance degradation in bulk data transmission to user's subjective assessment with 91 subjects. Moreover, from results of the experiment, we clarify how much throughput degradation compared with the usual network environment makes a user be felt dissatisfaction with. For example, our results can also apply to migration of

Table 1. Mean Opinion Score

MOS	Impairment
5	Imperceptible
4	Perceptible but not annoying
3	Slightly annoying
2	Annoying
1	Very annoying

network protocols. This is because minimizing users' dissatisfaction adopting an old protocol and keeping advantages of a new protocol are important.

The rest of the paper is organized as follows. In Section 2, we describe measurement methods to obtain user's subjective assessment. The details of our experimental environment are shown in Section 3. From our experiment, we discuss how much the degradation of throughput users can accept in bulk data transmission, and estimate a threshold of user's acceptable bandwidth in Section 4. Finally, we conclude our paper with future topics.

2 Methodologies for Measuring Subjective Assessment

We describe the details of measurement methods for user's subjective assessment. We also describe procedures to determine the threshold of user's acceptable bandwidth by using results of the measurements.

2.1 The Method of Successive Categories

The method of successive categories has two steps. First, subjects classify a stimulus into a certain number of categories that each of the categories is assigned an integer value, which is described as *rating scale method* [7]. The integer value is typically 5 through 1 in order of highly perceived quality of service, which is called as MOS (Mean Opinion Score). Here, the stimuli mean change of throughput in bulk data transmission. In our experiment, we define the five categories of impairment as shown in Table 1.

However, the obtained scores by rating scale method mean only a magnitude relation between scores, because MOS is an ordinal scale. Therefore, second, by using *the law of categorical judgment* [7], we translate values of MOS into interval scales. Moreover, the border lines of the interval scales between categories are obtained. In the interval scale, intervals between the score values mean differences of subject's senses directly. In order to estimate the threshold of the bandwidth in throughput degradation of bulk data transmissions, this method uses the interval scale translated from the MOS measured. The threshold estimated by this method is point at the intersection of a regression line of the translated MOS with the border line between Category 3 and Category 4, namely between MOS= 3 and MOS= 4.

2.2 The Constant Method

The constant method is one of psychometric methods which is regarded as the most accurate to estimate the threshold of stimulus. In the constant method, we need to select the number of stimuli, which is usually used from four to seven, by preliminary trials. Moreover, we also predict the threshold of stimulus at the same time for deciding values of stimuli. After the preliminary trials, we measure the subjective scores for each stimulus. Each stimulus is presented to a subject 10 times randomly. In our measurement, the subjective assessment is the alternative of “OK” or “NO”. Based on the measurement results, we calculate the rate of the number of “OK” responses to the number of all responses at each stimulus S , which rate is referred as $C_{OK}(S)$. Moreover, performing regression analysis to the distribution of $C_{OK}(S)$, we obtain the 50 percentile of the regression curve as a threshold of user’s satisfaction.

3 Experiment

We firstly show the network topology for our experiment. We also describe the details of our experiment to obtain the subjective assessments in bulk data transmission.

3.1 Experimental Network

Figure 1 shows the configuration of our experimental network. The network has a FTP server, a router, and a client machine which specifications are shown in Table 2. We control a bandwidth between the FTP server and the client machine at the router by using Dummynet [8]. A subject at the client machine downloads a data file from the FTP server, and assesses the quality of FTP service as a subjective assessment.

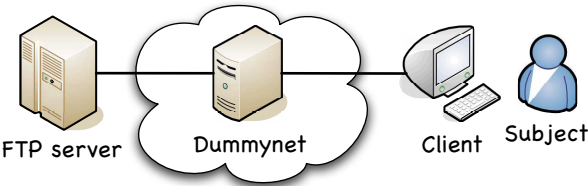


Fig. 1. Experimental network

Table 2. Machine specification

CPU	Pentium4 2.4 GHz
Memory	DDR400 512 MB
OS	Debian GNU/Linux 3.1 (kernel 2.4.26) or FreeBSD 5.4-Release
NICs	Intel PRO/1000 MT

3.2 Experimental Overview

In order to measure scores of the subjective assessment in throughput degradation in bulk data transmission, we firstly decide a criterial bandwidth and the size of a data file which a subject downloads in our experiment. A comparative evaluation of degradation of throughput is performed by comparing with the criterial bandwidth. Table 3 shows the configuration of our experiment, which measures the subjective scores in the case of throughput degradation in bulk data transmission. The size of a data file is decided from the software download ranking in Vector [9], which is one of famous software download sites in Japan. We set the median of the size of the software ranked in the top 10 as the size of the data file. Moreover, the value of the criterial bandwidth is set to the mean of the bandwidth of FTTH published at BNR [10], which is a benchmark site of link speed using by various ISP users.

Moreover, to decide the value of stimuli, we do a preliminary trial which is described in the constant method. In this experiment, the value of stimulus means the bandwidth restricted at the router. We set 7 as the number of stimuli. In the preliminary trial, 5 subjects firstly download the data file from FTP server at the criterial bandwidth on several occasions, and remember the speed of downloading the data file. Next, we measure the subjective assessment as the bandwidth is narrowing from the criterial value by 1 Mbps. Based on the preliminary trial, we predict 12 Mbps as a threshold of bandwidth. Thus, we set 7 stimuli of which the median is 12 Mbps, as shown in Table 3.

The number of subjects in this experiment is 91. They are Japanese males and females at ages between 20 and 24. The necessary time to complete the experiment was approximately 40 minutes. The procedure of our experiment is shown as follows.

1. A subject obtains the data file at 25 Mbps several times for acclimatizing the experimental network environment.
2. We change the bandwidth between the FTP server and the client machine to the one selected from the 7 stimuli at random.
3. The subject downloads from FTP server in the selected stimulus, assesses the quality of FTP service in both the constant method and the method of successive categories.

Each stimulus is selected 10 times randomly through the experiment. Therefore, the total number of the trials for each subject is 70 (7 stimuli $\times$ 10). During

Table 3. The configuration of our experiment

Size of data file	40 MB
Criterial bandwidth	25 Mbps
Stimuli	6, 8, 10, 12, 14, 16, 18 Mbps
# of trials	70 (10 times each stimulus)
# of subjects	91

downloading the data file, a progress bar is displayed to the subject. The subjective assessment of the constant method is either “OK” or “NO”. “OK” means that a subject feels satisfaction with the quality of that service. Moreover, in the method of successive categories, a subject scores the quality by selecting from the MOS values as shown in Table 1.

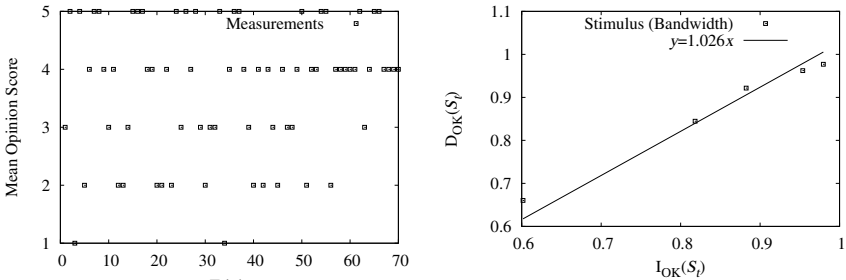
4 Results

We firstly show results of measurements in our experiment. Moreover, we discuss determination of the threshold of bandwidth for throughput degradation in bulk data transmissions. We also discuss how a user feels throughput degradation compared with the usual network environment.

4.1 The Overview of the Measurements

Figure 2(a) shows results of measurements assessed by Subject 5 through an experiment. In this figure, the horizontal axis is the trial number which is from 1 to 70, and the vertical one is the value of MOS. We use the measurements as shown above to estimate the threshold of subject’s satisfaction. In our experiment, the criterial bandwidth which should be compared with each stimulus is presented to the subjects only in initial step of our experiment. Therefore, the measurements may contain an error which is affected from stimulus presented previously. Namely, the subjects may assess the presented stimulus compared with the previous stimulus, not with the criterial bandwidth. Thus, we validate our measurements before analysis and estimation.

Figure 2(b) shows the difference of the subjective assessment in the constant method at several stimuli. The horizontal axis is the value of $I_{OK}(S_t)$, and the vertical one is the value of $D_{OK}(S_t)$. Let t and S_t denote the trial number range from 1 to 70 and the stimulus at trial t , respectively. We define $I_{OK}(S_t)$ as the ratio of the number of “OK” responses to the number of all responses when $S_t > S_{t-1}$. If $S_t < S_{t-1}$ is true, the ratio is defined as $D_{OK}(S_t)$. We perform regression



(a) The method of successive categories (b) Comparison of $I_{OK}(S)$ and $D_{OK}(S)$

Fig. 2. Measurements of Subject 5 and validation of measurement results

analysis and obtain the regression line, which is $y = 1.026x$. From this result, there is not the correlation between $I_{OK}(S_t)$ and $D_{OK}(S_t)$. Moreover, in the method of successive categories, we do statistical hypothesis testing for the MOS obtained. As a result, we found that the test with a significant level of 0.05 cannot reject the hypothesis that $I_{OK}(S_t)$ and $D_{OK}(S_t)$ are obtained from the same parent population. Thus, we found that in assessing the stimulus S_t , the subjects are not affected by the previous stimulus S_{t-1} . Therefore, we consider that the subjects assess the presented stimulus compared with the critical bandwidth.

4.2 Estimation of a Threshold of User's Satisfaction

We estimate a threshold of user's satisfaction for bulk data transmission, based on results of the measurements.

The Method of Successive Categories

We calculate the interval scales from the measured MOS. In order to obtain the interval scales, we apply the law of categorical judgment to all the measured MOS, which are classified into each stimulus. As a result, the interval scales translated from the measured MOS at each stimulus are shown in Fig. 3(a). Moreover, we also obtain the upper boundary of a category as 0.0 for MOS= 1, 0.9455 for MOS= 2, 2.1849 for MOS= 3, and 3.5349 for MOS= 4. From this result, in the interval scales, it is obvious that the distances between categories are not uniform. The regression line fitting to the translated interval scales of each stimulus is $y = 0.26x - 0.54$, which is shown as the dotted line in Fig. 3(a). The threshold of user's satisfaction is the x value of $y = 0.26x - 0.54$ when $y = 2.1849$. Therefore, we estimate that the threshold of user's satisfaction is approximately 10.46 Mbps. This result shows the subjects do not feel dissatisfaction with throughput degradation down to 52% compared with the critical bandwidth.

The Constant Method

We firstly calculate $C_{OK}(S)$ at each stimulus. Figure 3(b) shows the value of $C_{OK}(S)$ at each stimulus. We perform a regression analysis to the distribution of $C_{OK}(S)$ by using normal distribution. The regression curve obtained from this analysis is shown in Fig. 3(b) as the dotted line. The mean and the variance of this normal distribution are 6.71 and 14.52, respectively. Thus, the threshold of user's satisfaction obtained from this method is 6.71 Mbps, which is down to 73% compared with the critical bandwidth.

4.3 Discussion

From results of the estimation described above, we found that there is quite difference between the estimated values of two methods. We consider that the constant method underestimates the threshold compared with the method of successive categories, because the variance is extremely large. If an ISP restricts the bandwidth down to the threshold obtained by the constant method in the

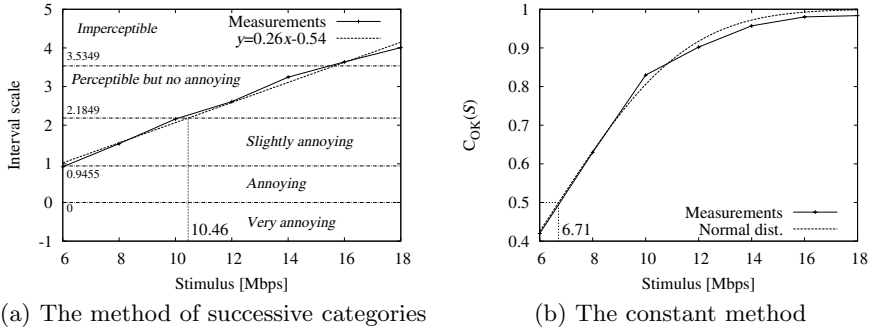


Fig. 3. Estimation of the threshold of user's satisfaction

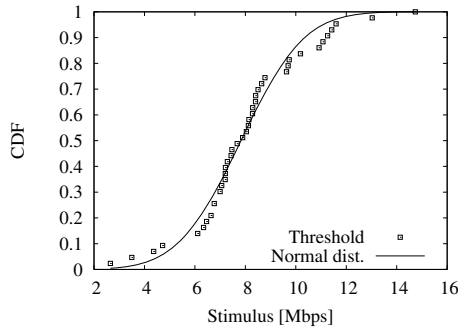


Fig. 4. Distribution of the threshold of each subject

case of competition of network resources between some users, the bandwidth allocated to the users may be restricted more than necessary. Moreover, from viewpoint of ISPs, such situation must be avoided, this underestimation may become a big problem. Although this method is easy and convenient to assess for a subject, we consider that an estimation result varies widely depending on preference or interest of a subject, especially near the value of the threshold. Therefore, in case of resource competition, an estimated threshold by the constant method is not preferable to restrict users' bandwidth, because it is necessary to treat carefully.

We next analyze the details of the estimated threshold of each subject. Figure 4 shows cumulative probability distribution of the threshold estimated from the measurements of each subject by using the constant method. The solid line in this figure is regression curve, which the mean and the variance of the normal distribution are 7.83 and 3.72, respectively. For example, from this figure, 80% of the subjects are not annoyed until the bandwidth degrades down to 9.5 Mbps, that is quite different from the estimated threshold (it is 6.71 Mbps) by the constant method described above. In contrast, the interval scales translated by the law of categorical judgment are increasing monotonically and stably

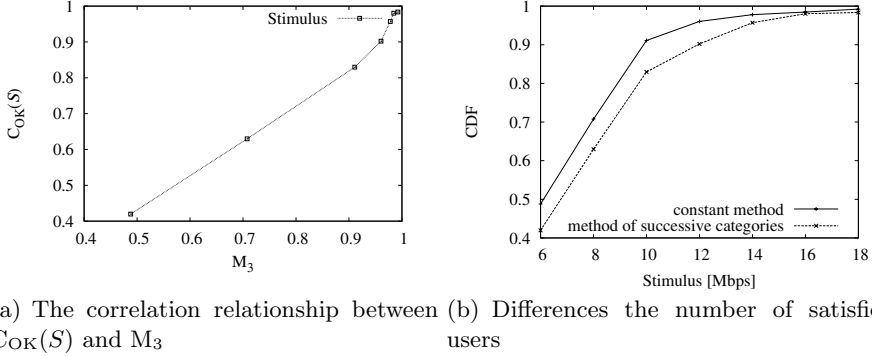


Fig. 5. Comparison of results of two methods

as shown in Fig. 3(a). This is because there is no significant fluctuation between assessments of subjects. Namely, fluctuations of the subjective assessment have non negligible impact to estimate the threshold of user's satisfaction which generates the difference of approximately 3.5 Mbps (14% compared with the criterial bandwidth) between two method. Therefore, by comparison with two methods we adopted, the method of successive judgment is more preferable for estimating the threshold of user's satisfaction in bulk data transmission. Moreover, our estimation results can be applied to the real network by using existing methods [4][5][6].

Moreover, we investigate the correlation of user's subjective assessment between two methods. Figure 5(a) shows correlation relationship between values of $C_{OK}(S)$ and M_3 at each stimulus. M_3 means the ratio of the number of responses with $MOS \geq 3$ to the number of all responses obtained from the subjects. Figure 5(b) shows difference of the number of satisfied users between two methods. From these result, we found that difference of two method is in range between 10 Mbps and 14 Mbps. We consider that in this region, subjects are unable to make a quick decision. In our experiment, we also found that the region is from the threshold obtained by the method of successive categories to the threshold plus 4 Mbps, which value is 16% of the criterial bandwidth.

5 Concluding Remarks

We firstly have measured subjective assessments by comparative evaluation in bulk data transmission by using psychological method. As a result, we have obtained 10.46 Mbps and 6.71 Mbps by using the method of successive categories and the constant method as thresholds of the bandwidth that the subjects begin to be annoyed, respectively. Furthermore, we also found that fluctuations of the subjective assessment have non negligible effect to the estimation result, that are occurred in range between the threshold and the threshold plus 4 Mbps, which value is 16% of the criterial bandwidth.

For future topics, in order to analyze more detail, we need to consider preferences or interests of a subject. This is because the subjective assessments is depended on kinds of contents, that contain music mp3 file and document file etc.. Moreover, we need to measure the subjective assessment in the case of different criterial bandwidth. We also need to consider any other network-level QoS such as loss rate and download time etc. to investigate effect of users' subjective assessment.

Acknowledgments

This work was partially supported by the Grant-in-Aid for Young Scientists (A) (No. 19680004) from the Ministry of Education, Culture, Sports, Science and Technology (MEXT) of Japan.

References

1. Braden, R., Clark, D., Shekner, S.: Integrated services in the Internet architecture: an overview. RFC 1633 (June 1994)
2. Blake, S., Black, D., Carlson, M., Davis, E., Wang, Z., Weiss, W.: An architecture for differentiated services. RFC 2475 (December 1998)
3. Ito, Y., Tasaka, S.: Quantitative assessment of user-level QoS and its mapping. *IEEE Transactions on Multimedia* 7(3), 572–584 (2005)
4. Tasaka, S., Ito, Y., Yamada, H., Sako, J.: A method of user-level QoS guarantee by session control in audio-video transmission over IP networks. In: *Proceedings of IEEE GLOBECOM 2006*, San Francisco, CA, pp. 1–6 (November 2006)
5. Kung, H.Y., Kuo, F.W.: Dynamic user-oriented QoS specification and mapping for multimedia differentiation services. *IJCSNS International Journal of Computer Science and Network Security* 6, 116–122 (2006)
6. Kulatunga, C.P., Malone, P., Foghlú, M.O.: Adaptive measurement based QoS in DiffServ networks. In: *IPS 2003* (February 2003)
7. Guilford, J.P.: *Psychometric Methods*. McGraw-Hill, New York (1954)
8. Rizzo, L.: Dummynet: a simple approach to the evaluation of network protocols. *ACM SIGCOMM Computer Communication Review* 27, 31–41 (1997)
9. Vector, <http://www.vector.co.jp/>
10. Broadband networking report, <http://www.musen-lan.com/speed/htmldata/>

A Space-Efficient Fair Packet Sampling Algorithm

Jin Zhang, Xiaona Niu, and Jiangxing Wu

National Digital Switching System Engineering and Technology Research Center (NDSC)
No. 7, Jianxue Street, Zhenzhou, China 450002
boost_zj@163.com, niuxiaona@ndsc.com.cn,
wujiangxing@ndsc.com.cn

Abstract. Due to the high-skewed nature of network flow size distributions, uniform packet sampling concentrates too much on a few large flows and ignores the majority of small ones. To overcome this drawback, recently proposed Sketch Guided Sampling (SGS) selects each packet at a probability that is decreasing with its current flow size, which results in better flow wide fairness. However, the pitfall of SGS is that it needs a large, high-speed memory to accommodate flow size sketch, making it impractical to be implemented and inflexible to be deployed. We refined the flow size sketch using a multi-resolution d-left hashing schema, which is both space-efficient and accurate. A new fair packet sampling algorithm which is named Space-Efficient Fair Sampling (SEFS) is proposed based on this novel flow size sketch. We compared the performance of SEFS with that of SGS in the context of flow traffic measurement and large flow identification using real-world traffic traces. The experimental results show that SEFS outperforms SGS in both application contexts while a reduction of 65 percent in space complexity can be achieved.

Keywords: Packet sampling, flow size estimation, multi-resolution sampling, d-left hashing.

1 Introduction

Accurate measurement and monitoring of network traffic is critical for various network management tasks such as usage accounting, resource planning, traffic engineering, pricing and billing, and anomaly detection. With the rapid growth of the Internet, network link speeds have become faster every year to accommodate more Internet users. Accurate monitoring of the traffic on such high speed links is a challenging data intensive problem. Packet sampling [1] serves as the de facto technology for reducing the amount of data that has to be processed when monitoring high speed links, which is now widely applied in many network traffic measurement and analysis applications such as per-flow traffic measurement [2-5] and traffic anomaly detection [9-11]. Random uniform sampling [1] is a straightforward and efficient way of packet sampling. Periodical sampling has a similar performance to uniform sampling while it is easy to be implemented in hardware, which is currently supported by equipment vendors (e.g., Cisco's NetFlow [2]).

However, due to the Zipfian nature of the Internet traffic - there are very few large flows but they contain the vast majority of the packets - uniform (or periodic)

sampling devotes too many resources on a few large flows and ignores the majority of the small ones. Thus uniform sampling is unsuitable for measuring and estimating flow specific information such as number of concurrent flows, flow size distribution and per-flow traffic. A recent work proposed Sketch Guided Sampling (SGS) [15] to overcome this pitfall. The key idea of SGS is making the sampling rate of a packet a decreasing function of its flow size. The flow sizes are estimated using a small (lossy) synopsis data structure (called a counting sketch). This way, SGS can significantly increase the packet sampling rate of the small and medium flows at slight expense of the large ones. The shortcoming of SGS is that it needs a large, high-speed memory to accommodate flow size sketch, making it impractical to be implemented and inflexible to be deployed. It is claimed by [15] that such a space complexity can be alleviated using statistic counters [22]. However, the flow size sketch of SGS needs support line-rate read operation, while the statistic counters proposed by [22] can only support line-rate update instead of line-rate reading. Furthermore, the estimation errors for small flows of SGS are unendurably large due to hash collisions.

In this paper, we presented a Space-Efficient Fair packet Sampling (SEFS) algorithm which is more space-efficient and accurate than SGS. Instead of using simple hash tables employed by SGS, we proposed a novel multi-resolution d-left hashing as the flow size sketch. The multi-resolution d-left hashing is composed of n independent d-left hashing counter arrays, the i -th counter array summarizes the size of each flows of the traffic which is sampled with a probability p_i ($p_i = P^i$ where P is the sample constant, $0 < P < 1$). Each counter array has a statistic range, the n counter arrays jointly cover the flow size domain. The multi-resolution d-left hashing sketch is more space-efficient and accurate than simple hashing sketch. We compared the performance of SEFS and that of SGS in the context of flow traffic measurement and large flow identification using real-world traffic traces. The experimental results show that SEFS outperforms SGS in both application contexts while a reduction of 65 percent in space complexity can be achieved.

The rest of this paper is organized as follows. Section 2 provides a brief survey of the related work. The concept of fair sampling is reviewed in section 3. Section 4 presents the novel multi-resolution d-left hashing sketch employed by SEFS. The estimation error as well as space and time complexity of multi-resolution d-left hashing sketch are also analyzed in this section. Section 5 explores the performance of SEFS and SGS in flow traffic measurement and large flow identification. Section 6 concludes this paper.

2 Related Work

Nick Duffield et al proposed weighted sampling method [12] to recover traffic size from sampled flow records. Adaptive sampling methods according to resource and traffic status are proposed in [13, 14]. A. Kumar et al presented the idea of fair packet sampling [15]. This paper is closely related to [15]. However, we proposed a novel flow size sketch that is more space-efficient and accurate than that employed in [15].

The idea of multi-resolution sampling is widely applied in various network traffic analysis applications, from estimating number of concurrent flows [18] to measuring per-flow traffic usage [19]. A. Kumar et al proposed a multi-resolution space-code Bloom filter (MRSCBF) to summarize traffic of each flows. However, MRSCBF is

inappropriate to serve as a flow size sketch because: 1) The time complexity of reading operation of space-code Bloom filter is $O(l)$ where l is the number of hash groups, which has a typical value of 32 or 64 [19]; 2) The flow size estimation error of space-code Bloom filter is orders of magnitude larger than that of d-left hashing [20].

Shijin Kong et al proposed a time-out Bloom filter [16] in order to sample more flows than uniform sampling. The key idea of [16] is recording the first packet of each flow. The pitfall of [16] is that it still can not ensure the fairness among flows, for the flow size information can not be recovered from sampled packets.

3 Fair Packet Sampling

Assume the size of flow F is f ($f \in [1, M]$). And assume sample each packet of F with probability p and total f' packets are selected. Then $\hat{f} = f'/p$ is the unbiased estimation of f , and the standard estimation error of f is $Std(\hat{f}) = \sqrt{(1-p)/fp} \approx 1/\sqrt{fp}$. Thus, the estimation error decreases with f under certain p . For large flows ($f \gg p^{-1}$ for example) the estimation error is relatively small while for small to medium flows ($f \ll p^{-1}$ for example) the estimation error is fairly large. Due to the zipfian nature of network flow sizes, uniform sampling incurs large error for majority flows thus it is inappropriate for many traffic measurement and analysis applications such as per-flow traffic measurement [2-5] and anomaly identification [9-11].

Fair packet sampling ensures the estimation error of any flow, $Std(\hat{f})$, equals to a constant ε regardless to its size. Assume we sample the i -th packet of flow F with a probability $P(i)$. The key idea of fair sampling is choosing $P(i)$ as a decreasing function of i . In this way, the packet sampling probability of large flows is decreasing with its size thus avoid be over sampled. It is easy to prove that when

$$P(i) = 1/(1 + \varepsilon^2(2i - 1)) \quad (1)$$

the fair sampling can be achieved [15], where ε is the expected standard estimation error. The following sampling function is also employed by [15] to compensate hash errors:

$$P(i) = 1/(1 + \varepsilon^2 i) \quad (2)$$

4 Multi-resolution d-Left Hashing Sketch

From formula (1) and (2) we are aware that the size of each flow is needed when perform fair sampling. A. Kumar et al used a simple hashing schema to estimate the size of each flow. This method is not space-efficient not accurate, especially for small flows. Assume there are N concurrent flows, and the maximum flow size is M , then the space complexity of SGS is $O(N \cdot \log_2 M)$.

We proposed a new flow size sketch using multi-resolution d-left hashing. The multi-resolution d-left hashing composed of n independent d-left hashing tables $C_1, C_2, \dots, C_n$, as demonstrated by Figure 1. C_i samples traffic with a probability $p_i = P^i$ (where P is the sample constant and $0 < P < 1$) and summarize the size each flow in the sampled traffic. The counting range of C_i is $[0, L+1]$. The state of $L+1$ means a

special ‘**overflow**’ state. The flow size counter remains unchanged when a overflow state is encountered when updating. The range of $[0, L]$ is called **effective counting range**. Assume the size counting value of flow F by $C_1, C_2, \dots, C_n$ are $c_1^F, c_2^F, \dots, c_n^F$ respectively, then the flow size of F are estimated by $\hat{f} = \text{estimate}(c_1^F, c_2^F, \dots, c_n^F)$ using certain estimation algorithm.

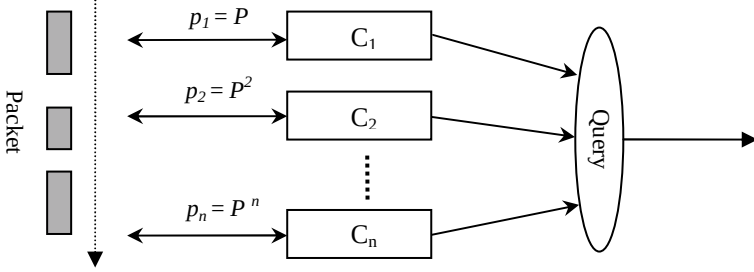


Fig. 1. Multi-Resolution d-left Hashing Sketch

Because the estimation procedure is a per-packet processing task, the estimation algorithm must be designed as simple as possible. Assume the size of F is f , then the standard estimation error of f when sampling with a probability p_i is given by [7]

$$\text{Std}(\hat{f}_i) = \frac{\sqrt{\text{Var}(\hat{f}_i)}}{f} = \frac{\sqrt{\frac{1}{p_i^2} \text{Var}(f'_i)}}{f} = \sqrt{\frac{1-p_i}{fp_i}} \quad (3)$$

It can be seen that the estimation is decreasing with the sampling probability p_i . Thus we choose the estimation algorithm as $\hat{f} = c_j^F / p_j$ where $j = \arg \min (c_i^F \in [0, L])$.

The estimation error is also affected by hash collision probability of d-left hashing, which is mainly determined by the length of element fingerprints [17]. We show that the effect of hash collision on estimation error can be ignored when using a 10 bits fingerprint. The detailed analytical and experimental results can be found in section 4.1.

4.1 Estimation Error Analysis

Before we analyze the estimation error of multi-resolution d-left sampling sketch, we give the definition effective coverage range.

Definition 1. Effective Coverage Range. Assume the effective counting range of C_i is $[0, L]$, then define $\omega_i = [\frac{1}{p_i}, \frac{L}{p_i}] = P^i[1, L]$ as the effective coverage range of C_i .

From formula (3) we know that $\forall f \in \omega_i$, the standard estimation error is $\sqrt{\frac{1-p_i}{fp_i}}$. Thus the upper bound of standard estimation error of flow size in ω_i is $\sqrt{1-p_i}$. However,

this bound can be further tightened due to the joint of several consecutive effective coverage ranges. If $\forall i \geq 1$, we have $\omega_i \cap \omega_{i+1} \cap \dots \cap \omega_{i+j-1} \neq \emptyset, \omega_i \cap \omega_{i+j} = \emptyset$, then we define j as **joint factor**. Lemma 1 give the upper bound of estimation error in each effective coverage range $\omega_i (i \geq j+1)$.

Lemma 1. Assume the joint factor is j , then $\forall f \in \omega_i (i \geq j)$, the estimation error of f is bounded by $Std(\hat{f}) \leq \sqrt{\frac{1-P^i}{LP}}$.

Proof: Assume the size of F is f , and f'_i packets are sampled after sample F with a probability p_i . Then $\hat{f}_i = f'_i / p_i$ is the unbiased estimator of f . Let $Std^{(i)}(f) = Std(\hat{f}_i)$ be the estimation error of f . We divide ω_i into j sections, that is, $\omega_i = \omega_i^1 \cup \omega_i^2 \cup \dots \cup \omega_i^j$, where $\omega_i^1 = [p_i^{-1}, Lp_{i-j+1}^{-1}]$, $\omega_i^2 = [Lp_{i-j+1}^{-1}, Lp_{i-j+2}^{-1}]$, ..., $\omega_i^j = [Lp_{i-1}^{-1}, Lp_i^{-1}]$. Thus

$$Std^{(i-j+1)}(f) \leq Std^{(i-j+1)}(p_i^{-1}), \forall f \in \omega_i^1 \quad (4)$$

$$Std^{(i-j+s)}(f) \leq Std^{(i-j+s)}(Lp_{i-j+s-1}^{-1}), \forall f \in \omega_i^s, 2 \leq s \leq j \quad (5)$$

substitute formula (3) into formula (4) and formula (5) respectively, thus we have:

$$Std^{(i-j+1)}(f) \leq \sqrt{P^{j-1}(1-p_{i-j+1})}, f \in \omega_i^1 \quad (6)$$

$$Std^{(i-j+s)}(f) \leq \sqrt{\frac{1-p_{i-j+s}}{LP}}, f \in \omega_i^s, 2 \leq s \leq j \quad (7)$$

from formula (6) and (7) we know $\forall f \in \omega_i$, the maximum value of $Std(\hat{f})$ is

$\max\left(\sqrt{P^{j-1}(1-p_{i-j+1})}, \sqrt{\frac{1-p_i}{LP}}\right)$. Because $j \geq 2$, thus $p_{i-j+1} > p_i$, and

$$\sqrt{P^{j-1}(1-p_{i-j+1})} / \sqrt{\frac{1-p_i}{LP}} < \sqrt{P^{j-1}(1-p_i)} / \sqrt{\frac{1-p_i}{LP}} = \sqrt{LP^j} \quad (8)$$

For the joint factor is j , thus $P^{1-j} < L < P^{-j}$, and we have $\sqrt{LP^j} < 1$, thus

$$\sqrt{P^j(1-p_{i-j+1})} / \sqrt{\frac{1-p_i}{LP}} < 1 \quad (9)$$

Then, $\forall f \in \omega_i$, the maximum value of $Std(\hat{f})$ is $\sqrt{\frac{1-P^i}{LP}}$. ■

Theorem 1. Assume the joint factor is j , then $\forall f \in [1, p_j^{-1}]$, the standard estimation error of multi-resolution d-left hashing is bounded by $\sqrt{(1-P)/P}$; $\forall f \in [p_j^{-1}, +\infty)$, the standard estimation error of multi-resolution d-left hashing is bounded by $1/\sqrt{LP}$.

Theorem 1 is derived under the assumption that no hash collision was occurred. But when the element fingerprints are sufficient long, the effect of hash collisions on estimation error can be ignored. Figure 2 shows the estimation error of multi-resolution d-left

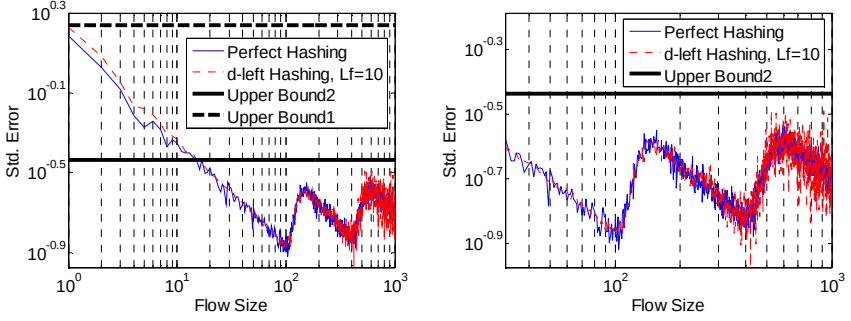


Fig. 2. The Effect of Hash Collisions on Estimation Error

hashing when $L = 30$, $P = 0.25$. The parameters of d-left hashing are setting as follows: the number of parallel selection is 4, the mean bucket depth is 4 and the extra bucket depth is 1. The element fingerprints length is 10. From figure 2 we can see that the effect of hash collision can be ignored under these settings.

4.2 Space Complexity Analysis

The total space occupied by multi-resolution d-left hashing sketch is the sum of the space occupation of each d-left hashing counter array. Assume the maximum flow size is M , the effective counter range of each d-left hashing is $[0, L]$, then the total number of d-left hashing counter array layer is

$$n = \lceil \log_p L - \log_p M \rceil \quad (10)$$

The space complexity of multi-resolution d-left hashing is given by Theorem 2.

Theorem 2. Assume the flow size attributes to a zipfian distribution with parameter z in range $[1, M]$. Then the space complexity of multi-resolution d-left hashing is

$$O(N'), \text{ where } N' = N\theta \sum_{i=1}^n \sum_{j=1}^M \frac{1 - (1 - P^i)^j}{j^z}, \text{ and } n \text{ is given by formula (10).}$$

Proof: Assume there are total N concurrent flows, and the number of flows summarized by C_i is N_i . Let $N' = \sum_{i=1}^n N_i$. For any flow of size f , the probability that at least one packet is sampled when sampling each of its packet with probability p is $1 - (1 - p)^f$, thus we have

$$N_i = \sum_{j=1}^M N \cdot P(F = j) \cdot (1 - (1 - p_i)^j) \quad (11)$$

where $P(F = j)$ equals to the probability that for any flow, its size is $j (1 \leq j \leq M)$. From the zipfian assumption we know that

$$P(F = j) = \frac{\theta}{j^z} \quad (12)$$

where $\theta = \left(\sum_{j=1}^M j^{-z} \right)^{-1}$ is the normalized constant. Thus we have

$$N' = N \theta \sum_{i=1}^n \sum_{j=1}^M \frac{1 - (1 - P^i)^j}{j^z} \quad (13)$$

■

4.3 Time Complexity Analysis

We use the number of memory accesses for each packet as the time complexity metrics of multi-resolution d-left hashing sketch. For each d-left hashing counter array C_i , the expected number of memory access on each packet arrival is one read operation and p_i update operation. The update operation can be regarded as one read and one write. Because each d-left hashing counter array has a counting upper bound of $L + 1$, thus the expected number of memory access time due to update operation per packet is less than $2p_i$. The total number of memory access per packet is less than

$$O\left(n + \frac{2P(1 - P^n)}{1 - P}\right), \text{ where } n \text{ is given by formula (10).}$$

5 Performance Evaluation

The sampled packets can be used for various traffic analysis applications such as flow size distribution estimation, per-flow traffic accounting, large flow identification and accounting as well as traffic classification. For the sake of comparison, in this paper we explored the performance of SEFS in the context of two typical traffic analysis applications, that is, per-flow usage accounting and large flow identification [15]. Because recent work of [15] has compared the performance of SGS and uniform sampling, we compare the performance of SEFS to that of SGS only.

The performance metrics of per-flow usage accounting is **standard estimation error (Std. Error)**. Assume there are total N concurrent flows. The size of flow

$F_i (1 \leq i \leq N)$ is $f_i (1 \leq f_i \leq M)$. The estimation value of f_i is $\hat{f}_i$. Let $e_i = \frac{|\hat{f}_i - f_i|}{f_i}$,

$E_j = \{e_i \mid f_i = j\} (j \in [1, M])$, then the Std. Error is defined as $\bar{e}_j = \frac{1}{|E_j|} \cdot \sum_{e_i \in E_j} e_i$.

The performance metrics of large flow identification are **hit ratio** and **false positive ratio**. Define all flows whose size is greater than T as large flow. Assume the set of large flows is F^T , the set of ‘detected’ large flows is $\hat{F}^T$, then the hit ratio is defined as $HR = \frac{|F^T \cap \hat{F}^T|}{|\hat{F}^T|}$, and the false positive ratio is defined as $FPR = \frac{|\hat{F}^T - (F^T \cap \hat{F}^T)|}{|\hat{F}^T|}$.

Furthermore, we are also interested in the mean estimation error of the size of large flows.

We use real-world traffic traces collected on OC-48 links published by NLANR. We used the first and the second minute traffic data of IPLS-KSCY-20020814-090000 (named trace1 and trace2 respectively). The flows are identified by

well-known five-tuple. Trace1 contains total 45,807 flows, with a maximum size of 112,008. Trace1 contains total 43,347 flows, with a maximum size of 76,954. We set $\varepsilon = 0.5$, $P = 1/16$, $L = 126$ throughout our experiment.

5.1 Per-Flow Usage Accounting

Figure 3 shows the Std. Error of per-flow size estimation of both SEFS and SGS. From Figure 3 we can see that for small flows, the estimation error of SEFS is significantly less than that of SGS while for large flows, the estimation errors of SEFS and SGS are similar.

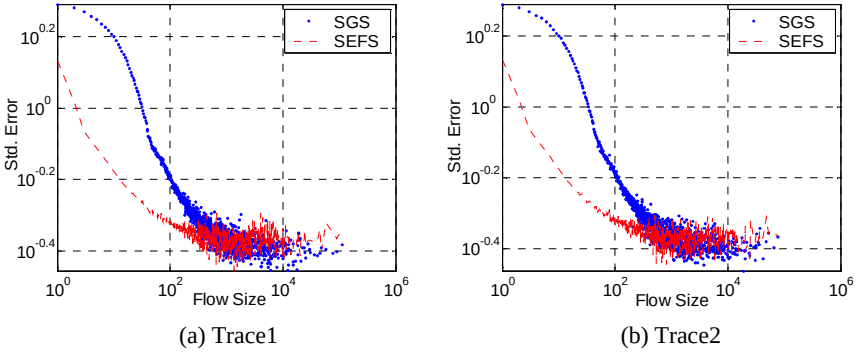


Fig. 3. Std. Error of Per-Flow Size Estimation

The per-flow space occupation of SEFS is a 10 bits fingerprint and a 7 bits counter. From formula (13) we know the fraction of flows sampled by multi-resolution d-left hashing sketch is 0.35. Then the per-flow space occupation of SEFS is less than 6 bits. However, the per-flow space occupation of SGS is 17 bits. Thus the space complexity of SEFS is 35 percent to that of SGS. This ratio can be further reduced when the maximum flow size increases.

5.2 Large Flow Identification

Table 1 and table 2 are performance comparisons of large flow identification of SEFS and SGS using Trace1 and Trace2 respectively. From Table 1 and Table 2 we can see that SEFS outperforms SGS remarkably, especially in sense of low False Positive Ratio.

Table 1. Performance of large flow identification of SEFS and SGS, using Trace1

	SGS			SEFS		
	$T = 8$	$T = 16$	$T = 32$	$T = 8$	$T = 16$	$T = 32$
Hit Ratio	0.5908	0.7106	0.8258	0.8029	0.7801	0.8239
False Positive Ratio	0.2715	0.3514	0.4503	0.0854	0.0427	0.0775
Mean Std. Error	0.4817	0.4757	0.4651	0.4411	0.4406	0.4387

Table 2. Performance of large flow identification of SEFS and SGS, using Trace2

	SGS			SEFS		
	$T = 8$	$T = 16$	$T = 32$	$T = 8$	$T = 16$	$T = 32$
Hit Ratio	0.5892	0.7127	0.8171	0.8060	0.8038	0.8330
False Positive Ratio	0.2599	0.3379	0.4261	0.0768	0.0583	0.0741
Mean Std. Error	0.4857	0.4793	0.4695	0.4417	0.4412	0.4389

6 Conclusions

Packet sampling is inevitable to various real-time traffic measurement and analysis tasks on high-speed backbone networks such as per-flow usage accounting, large flow identification, anomaly detection, and traffic classification. Fair packet sampling can obtain much information than uniform packet sampling with the sampling rate. In this paper we proposed a space-efficient fair packet sampling (SEFS) algorithm. The key contribution of SEFS is a novel flow size sketch that is both space-efficient and accurate. The performance of SEFS is compared with that of existing fair packet sampling algorithm (SGS) in respect of per-flow usage accounting and large flow identification. The experimental results show that SEFS outperforms SGS in both application contexts while a reduction of 65 percent in space complexity can be achieved. The performance of SEFS in the context of other traffic monitoring and analysis application areas such as anomaly detection and traffic classification will be explored in future works.

References

1. Claffy, K.C., Polyzos, G.C., Braun, H.-W.: Application of Sampling Methodologies to Network Traffic Characterization. In: Proc. ACM SIGCOMM (1993)
2. Cisco System White Paper. NetFlow Services Solutions Guide
3. Estan, C., Varghese, G.: New directions in traffic measurement and accounting. In: Proc. ACM SIGCOMM (August 2002)
4. Kodialam, M., Lakshman, T.V., Mohanty, S.: Runs bAsed Traffic Estimator (RATE): A simple, Memory Efficient Scheme for Per-Flow Rate Estimation. In: IEEE Proceedings of INFOCOM (2004)
5. Raspall, F., Sallent, S., Yufera, J.: Shared State Sampling. In: Proc. ACM Internet Measurement Conference (2006)
6. Hohn, N., Veitch, D.: Inverting Sampled Traffic. In: ACM Internet Measurement Conference (2003)
7. Duffield, N., Lund, C., Thorup, M.: Properties and Prediction of Flow Statistics from Sampled Packet Streams. In: Proc. ACM Internet Measurement Conference (2002)
8. Barakat, C., Iannaccone, G., Diot, C.: Ranking flows from sampled traffic. In: Proceedings of the 2005 ACM conference on Emerging network experiment and technology, pp. 188–199 (2005)
9. Brauckhoff, D., Tellenbach, B.: Impact of Packet Sampling on Anomaly Detection Metrics. In: ACM Internet Measurement Conference (2006)

10. Mai, J., Chua, C.-N.: Is Sampled Data Sufficient for Anomaly Detection. In: ACM Internet Measurement Conference (2006)
11. Chen, G., Gong, J., Ding, W.: A Real-Time Anomaly Detection Model Based on Sampling Measurement in a High-Speed Network. *Chinese Journal of Software* 3 (2003)
12. Duffield, N., Lund, C., Thorup, M.: Learn More, Sample Less: Control of Volume and Variance in Network Measurement. *IEEE Transactions on Information Theory* 51(5) (May 2005)
13. Estan, C., Keys, K.: Building a Better NetFlow. In: Proc. ACM SIGCOMM (2005)
14. Wang, J., Yang, J., Zhou, H., Xie, G., Zhou, M.: Adaptive Sampling Methodology in Network Measurements. *Chinese Journal of Software* 15(8) (2004)
15. Kumar, A., Xu, J.J.: Sketch Guided Sampling – Using On-Line Estimates of Flow Size for Adaptive Data Collection. In: IEEE Infocom 2006 (2006)
16. Kong, S., He, T., Shao, X., Li, X.: Time-out Bloom Filter: A New Sampling Method for Recording More Flows. In: Chong, I., Kawahara, K. (eds.) ICOIN 2006. LNCS, vol. 3961, pp. 590–599. Springer, Heidelberg (2006)
17. Bonomi, F., Mitzenmacher, M., Panigrahy, R., Singh, S., Varghese, G.: An Improved Construction for Counting Bloom Filters. In: European Symposium on Algorithms (2006)
18. Estan, C.: Bitmap Algorithms for Counting Active Flows on High Speed Links. In: Proceedings of the 3rd ACM SIGCOMM conference on Internet measurement, pp. 153–166 (2003)
19. Kumar, K., Xu, J., Jia, W., Spatschek, O., Li, L.: Space-Code bloom filter for efficient per-flow traffic measurement. In: Proc. of the INFOCOM 2004 (2004)
20. Zhang, J.: Performance Evaluation and Comparison of Three Counting Bloom Filter Schemes. Technical Report (2008)
21. NLANR. Abilene-I data set,
<http://pma.nlanr.net/Traces/long/ipls1.html>
22. Shah, D., Iyer, S., Prahakar, B., McKeown, N.: Maintaining statistics counters in router line cards. *IEEE Micro*. 22(1), 76–81 (2002)

Understanding IPv6 Usage: Communities and Behaviors

Shaojun Huang¹, Changqing An², Hui Wang², and Jiahai Yang²

¹ Network Research Center, Tsinghua Univ. Beijing China

² Tsinghua National Laboratory for Information Science and Technology
huangsj02@mails.thu.edu.cn, {acq,hwang,yang}@cernet.edu.cn

Abstract. The most important challenge faced by current IPv6 networks is to attract more users. Our data analysis of traces from exit points of CERNET2 shows that the traffic in this network has increased a lot during last five months. In this paper, we try to find the incentives of IPv6 users by identifying host communities and studying their properties. We reveal that popular services on CERNET2 are Web, FTP and VOD. FTP and VOD are the killer applications in terms of traffic volume while Web produces the largest number of flows. By analyzing behaviors of end hosts, we discover that hosts form many communities. These communities have different flow-level topologies and various interests in services. Using IPv6 prefixes assignment data, we perform demographic study of IPv6 usage. By presenting CERNET2 usage from various perspectives, we believe this study provides insight into further deployment of IPv6.

1 Introduction

The China Next Generation Internet (CNGI) project aims to establish IPv6-oriented next generation Internet infrastructure. It is comprised of six national or regional backbone core networks which are interconnected through two Internet eXchange points located in Beijing and Shanghai respectively. CERNET2 is one of the six CNGI backbone core networks. In its infancy, the most important challenge faced by current IPv6 network is to attract more users. Recent statistics of traffic at exit points of CERNET2 shows that the traffic intensity in this network has increased dramatically during last five months.

It is important to understand and characterize the usage of IPv6 so as to provide insight into future IPv6 deployment. But until today, most works are on IPv4 networks. On one hand, researchers analyze Internet performance and work on traffic characterization on IPv4 networks. Though IPv6 resembles IPv4 in a number of ways, we are not sure all results obtained from IPv4 can apply to IPv6 networks. On the other hand, there are few works about IPv6 usage. Most works we know are deployment reports or simple analysis of traffic volume. This is insufficient for future IPv6 deployment and planning. An in-depth analysis of current IPv6 usage and user behaviors is instrumental in future IPv6 traffic engineering and network design.

In this work, we present a systematic analysis of CERNET2 usage on various aspects. We introduce Uncertainty Level (*UL*) and propose a heavy hitter identification method called UL partition. We identify popular services by our heavy hitter identification method. By identifying port numbers that generate most traffic, we discover three most popular services on CERNET2. Using a labeling technique, we identify each flow and reveal corresponding network services. We confirm that most flows are generated by those popular services. We reveal that VOD is the killer application of CERNET2. Using several community finding methods, we group hosts into communities at flow level and analyze their behaviors. We find that some communities are in star topology and some form cliques. There are also end hosts that belong to multiple communities. We also present demographic study of CERNET2 usage.

The remainder of this paper is organized as follows; in section 2, we describe related work; later in section 3, we depict a general profile of CERNET2 and traces used in the research. In section 4, we describes in detail our research methodology. Next in section 5, we reveal popular services and identify killer applications. In section 6, we identify host communities from *IPs* and *Ports* and analyze community properties and interests. Demographic study is presented in section 7 and possible incentives are also discussed. Finally, we summarize our work and conclude with section 8.

2 Related Work

To understand the Internet, researchers mainly focus on traffic characterization and topology measurement. On topology studies, much research has been done on Internet topology at router-level. Lun Li et al studied router-level and AS-level topology of the Internet in terms of network connectivity[1]. Lots of works[2] revealed heavy-tail distribution in node degree and power-law distribution. Based on the discovered characteristics, they also built various topology generators. On traffic characterization, Crotti et al [3] classify traffic using statistical fingerprinting. Xu and Zhang[4] profile Internet traffic using data mining technique. On P2P, many researchers work on the identification of P2P traffic[5]. Karagiannis et al in their research[6] identify P2P traffic based on communication patterns of P2P nodes, social behavior of hosts and community of hosts. Dario et al[7] classify Skype traffic using various Bayesian classifier. Lihua et al introduce a novel concept *flowset*[8].

On social network analysis, there are works on topics such as network properties, identification of communities, relationship among participants, etc. In recent years, researchers focus on mining the Web and blog community[9][10]. Various community metrics are proposed and evaluated, such as Normalized-Cut[11], rawColumn [12], etc. Researchers shift their focus to relationships among network nodes from Power-law distribution of node degrees and the small world property.

3 Dataset Description

CERNET2 connects 25 GigaPOPs distributed in 20 cities across the country. CERNET2 has peered with Abilene/Internet2, TEIN2, and APAN. It has also interconnected with other backbone core networks of CNGI. Till now, about top 100 universities and other academic and research institutions in China have been connected to CERNET2 with various access technologies, including dual-stack native IPv6, IPv6 over IPv4 tunneling technology, IPv4 over IPv6 tunneling technology, etc. Traces are collected from exit points of CERNET2. Traces are stored on a manner of 10 minutes per file. Packet header traces of five months are collected. We focus on allocated IPv6 prefix only. Instead of studying behaviors of individual host, we work on prefixes of specific length, typically 32bits and 48bits.

4 Methodology

Suppose X is a random variable with N possible values $\{x_i\}, i \in [1, N]$ and a probability distribution $\{p(x_i)\}$, information theory quantifies its amount of information, i.e., entropy as $H(X) = -\sum p(x_i)\log(p(x_i))$. Suppose we observe X for m times and x_i appears m_i times, we get an empirical probability distribution of X as $\hat{p}(x_i) = m_i/m$. Xu and Zhang[4] define relative uncertainty (RU) as follows

$$RU(X) = \frac{H(X)}{\log(\min(N, m))} = \frac{\sum -\hat{p}(x_i)\log(\hat{p}(x_i))}{\log(\min(N, m))} \quad (1)$$

RU can be used to measure the uniformity of random variable X .

We define probability space $(\Omega_{flow}, \mathcal{F}_{flowset}, P)$ where Ω_{flow} is the set of flows defined by the five tuple $(srcIP, srcPort, dstIP, dstPort, proto)$ and $\mathcal{F}_{flowset}$ is a σ -algebra over Ω_{flow} . We refer to the following *flowsets* in our study as $SDIP = \{(srcIP, *, dstIP, *, *)\}$, $SDPO = \{(*, srcPort, *, dstPort, *)\}$ where $*$ denote arbitray value. Given a threshold $\beta > 0$, heavy hitters of $\mathcal{F}_{flowset}$ belong to the set $Y = \{x|p(x) > \beta\}$ and the rest belong to subset $Z = \{x|p(x) \leq \beta\}$. We define uncertainty level (UL) of Y in terms of X as

$$UL(Y|X) = -\log\left(\frac{RU(Y)}{RU(Z)}\right) \quad (2)$$

UL can be used to measure the outstanding level of a set of heavy hitters. We define UL partition as finding the set of Y such that $UL(Y|X) \geq \epsilon$ where $\epsilon \geq 0$. We use this to extract heavy hitters.

After identifying heavy hitters, we formulate the traffic classification problem as a function from $\mathcal{F}_{flowset}$ to a set of labels, say $\{\text{web, p2p, ftp, ...}\}$. For example, a label $\{\text{p2p, bt}\}$ means a p2p flow that is BT. We compute the classification function iteratively using algorithm 1 which will be described in detail in section 5.

5 Service Behaviors

Both TCP and UDP protocols use port numbers to identify sending and receiving applications; thus, port numbers can be used as identifiers of services. Let $f_{srcPort}$ be $\{f | f \in \Omega_{flow} \wedge \forall x, y \in f, x.srcPort = y.srcPort\}$ and $f_{dstPort}$ is defined in a similar way, we use UL partition to extract heavy hitters in terms of percentage of bytes, packets and flows, respectively. We simply choose ϵ to be zero. The result is demonstrated in Table 1.

Table 1. Typical port heavy hitters, TCP and UDP protocol

Typical port heavy hitters, TCP protocol		
	srcPort	dstPort
bytes	80 1054 1068 1048 1035 1044 1034	20 554 21627 48515 44805 9543
packets	1054 1068 1048 1035 1044 1034 1106	20 554 21627 48515 44805 9543 37864
flows	80 1025 1029 1027 1026 1028	554 80 21 20 1755 1021
Typical port heavy hitters, UDP protocol		
	srcPort	dstPort
bytes	53 5004 62972 1177 1063 7001	5005 32777 42114 11008 32770 39953
packets	53 5004 62972 1177 1063 7001	1057 5005 42114 32777 11008 39953 32770
flows	53 6881 3540 32769 49152 1031	30267 38225 5005 11008 24540 62374 47407

In order to understand service behaviors, we visualize $(srcPort, dstPort)$ pairs in Figure 1. An obvious characteristic is that clusters form lines perpendicular to axes. We can see in Figure 1(a) that port pairs tend to form lines that are perpendicular to srcPort axis. Clusters formed within port range near 50000 cannot be identified directly. On the contrary, for UDP protocol, most clusters in Figure 1(b) are perpendicular to dstPort axis. Top 200 UDP port pairs plotted as triangles in Figure 1(b) also tend to be perpendicular to srcPort axis.

Besides well known ports, there are still many flows that use dynamic ports. For these flows, we cannot determine their associated service directly. So we identify them using the following method. Given $(\Omega_{flow}, \mathcal{F}_{flowset}, P)$, let S be a set of labels, we call function $Cl : \mathcal{F}_{flowset} \mapsto 2^S$ flow classification function. We select

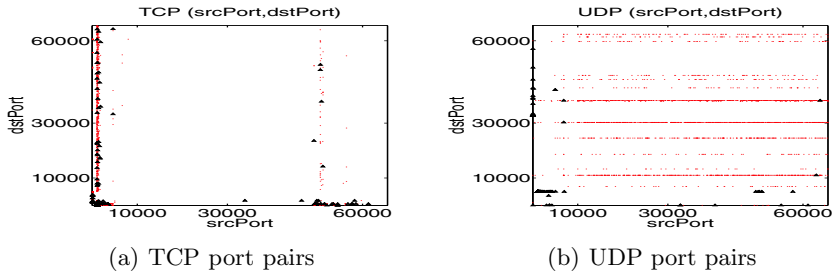


Fig. 1. $(srcPort, dstPort)$ pairs for TCP and UDP protocol. Top 200 flows are plotted as black triangles in the figure.

$S = \{\text{server}, \text{client}, \text{ftp}, \text{ssh}, \text{telnet}, \text{web}, \text{dns}, \text{mms}, \text{email}, \text{snmp}\}$ and label each flow l with $S_l \subseteq S$. For example, if a flow is labeled $\{\text{server}, \text{mms}, \text{web}\}$, we say that it is a VOD flow since MMS servers use http protocol to transmit control sequences; if an IP is labeled $\{\text{client}, \text{web}, \text{mms}, \text{ssh}\}$, we say that this host is a client of web, MMS and SSH. At the beginning, we determine a small subset $Cl_s \subseteq Cl$ by labeling heavy hitters, e.g., port 20 indicates a FTP traffic. We label $\mathcal{F}_{flowset}$ iteratively using algorithm 1 and use the result as a proximation of Cl . We give an example of algorithm 1 here. Suppose we find a heavy hitter, say TCP port 80, we are sure that 80 is used by Web servers, so we label all flows whose *srcPort* is 80 $\{\text{web}\}$ and corresponding *srcIP* $\{\text{web}, \text{server}\}$. Suppose an IPv6 address IP_1 is labeled $\{\text{ftp}, \text{server}\}$, all flows that satisfy conditions either *srcIP* or *dstIP* be IP_1 will be labeled $\{\text{ftp}\}$ and corresponding *dstIP* or *srcIP* will be labeled $\{\text{ftp}, \text{client}\}$.

Algorithm 1. Computation of $Cl(f)$

```

 $Cl(f) \leftarrow \bigcup (Cl(f), Cl(srcPort), Cl(dstPort)) - \{\text{server}, \text{client}\}$ 
if  $\text{server} \in Cl(srcIP)$  then
     $Cl(f) \leftarrow (Cl(f) \cup Cl(srcIP)) - \{\text{server}, \text{client}\}$ 
end if
if  $\text{server} \in Cl(dstIP)$  then
     $Cl(f) \leftarrow (Cl(f) \cup Cl(dstIP)) - \{\text{server}, \text{client}\}$ 
end if
if srcPort is well known then
     $Cl(srcIP) \leftarrow Cl(srcIP) \cup \{\text{server}\}$ 
     $Cl(dstIP) \leftarrow Cl(dstIP) \cup \{\text{client}\}$ 
end if
if dstPort is well known then
     $Cl(dstIP) \leftarrow Cl(dstIP) \cup \{\text{server}\}$ 
     $Cl(srcIP) \leftarrow Cl(srcIP) \cup \{\text{client}\}$ 
end if
 $Cl(srcIP) \leftarrow Cl(srcIP) \cup Cl(f)$ 
 $Cl(dstIP) \leftarrow Cl(dstIP) \cup Cl(f)$ 

```

We examine both TCP and UDP traffic using algorithm 1. By cross-examining every flow using the labeling technique, we discover that random port numbers are due to FTP passive mode traffic. A random port pair is negotiated and used in FTP passive mode. We find that most TCP traffic is Web, FTP and a little VOD traffic. Most UDP traffic is MMS traffic. Other types of traffic appear randomly and are small in flow size. Another cluster perpendicular to *dstPort* can be explained by the characteristics of MMS traffic that use separate ports for inbound and outbound.

Properties of *srcPort* and *dstPort* are presented in Figure 2. Figure 2(a) and 2(c) demonstrate the average packet size for *srcPort* and *dstPort* dimension. Figure 2(b) and 2(d) demonstrate number of packets transmitted over a specific port for *srcPort* and *dstPort* dimension. By looking into packets transmitted over *srcPort* and *dstPort* dimensions, we discover that most packets are transmitted by ports smaller than 10000.

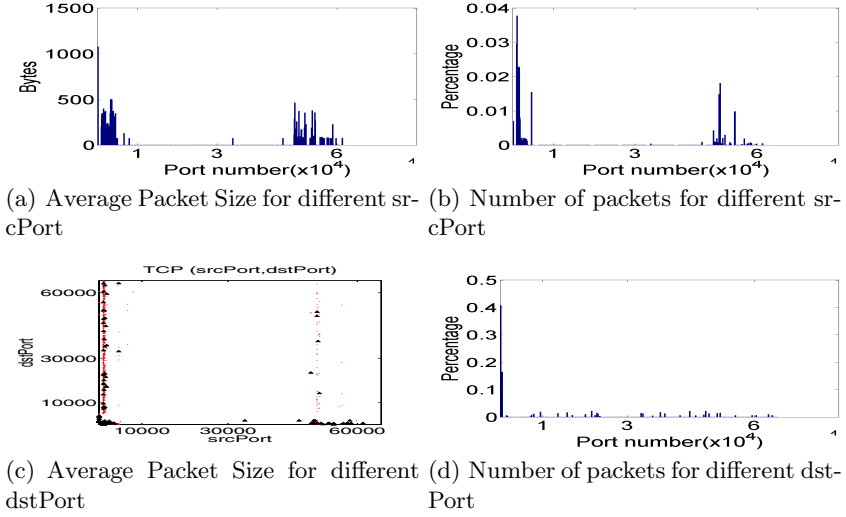


Fig. 2. Packet count and average packet size of srcPort and dstPort dimensions

6 Host Behaviors

In this section we study host behaviors by identifying host communities. Hosts inside a community tend to communicate with each other more than with outsiders. We also present community properties and reveal community interests.

6.1 Host Communities

We model IPv6 network as a weighted undirected graph $G = \langle V, E, \omega \rangle$, where V denotes the set of hosts observed, $E = \{(i, j) | i, j \in V \text{ and there are flows between } i \text{ and } j\}$ and $\omega(i, j)$ equals to traffic volume between i and j . IPv6 address space is divided into two disjoint subsets according to prefixes, one called CERNET2 and the other called non-CERNET2. We define communication matrix $CM = (a_{i,j})$ where $i \in \text{CERNET2}$, $j \in \text{non-CERNET2}$ and $a_{i,j}$ equals to $\omega(i, j)$.

A community C of G is a subgraph that has more internal edges, i.e., $\{(i, j) | i, j \in V(C)\}$ contains more edges than $\{(i, j) | i \in V(G - C), j \in V(C)\}$. A practical criteria of identifying communities is to minimize the number of edges connecting hosts from different communities. One of such algorithm is Normalized-Cut[11]. We identify host communities using Normalized-Cut algorithm. Figure 3 is a partial representation of the Communication Matrix. A dot in the graph means that there are flows between the two. Figure 3 shows a possible classification of hosts into five communities. Dots of the same color and shape indicate that corresponding hosts are in the same community.

Two examples of host communities are demonstrated in Figure 5, one in star topology and the other a clique. Various types of topology have different implications. A star topology implies that such community is a client/server community

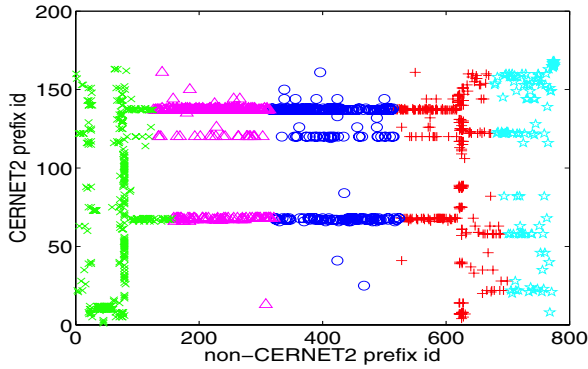
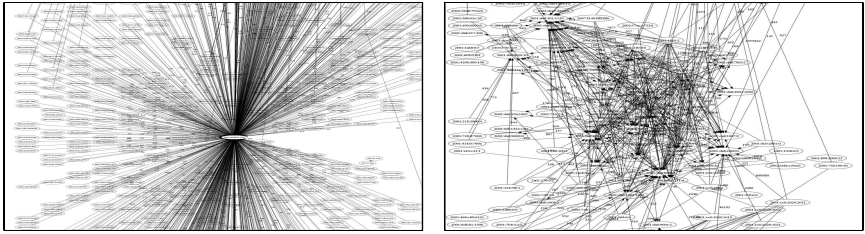


Fig. 3. Communities identified by clusters of *SDIP*



(a) A host community with star topology (b) A host community that is a clique

Fig. 4. Two examples of identified host communities

and the host with largest degree is likely to be the server. Another implication in star topology is that there are hosts that only generate flows to a specific server. If a user browses the Internet and send email simultaneously, we are suppose to see that such a host will have many flows with many servers at the same time.

6.2 Community Interests

In this section, we present interests of communities identified in previous section. Community interest describes what types of network services a community uses mostly. For example, if a set of hosts use BitTorrent protocol to share files with each other, we say that its community interest is the P2P application. Since most IPv6 traffic can be classified according to port numbers, we are able to identify community interests by port numbers that generate most traffic. We discover community interests by identifying port heavy hitters in each community. Traffic data is partitioned into five groups. For each group, we perform UL partition to identify port heavy hitters and find out corresponding well-known ports. The result is shown below in Table 2. An example of communities and their interests can be seen in Figure 5.

Table 2. Discovering Community Interests

Community	1	2	3	4	5
TCP port	554 1755 80	20 21	20 21 554 1755 80	80	554 1755 80
UDP port	5005	3540 30627 38225	5005	N/A	1755
Interests	Video On Demand	FTP	FTP, Video On Demand	Web	Video On Demand

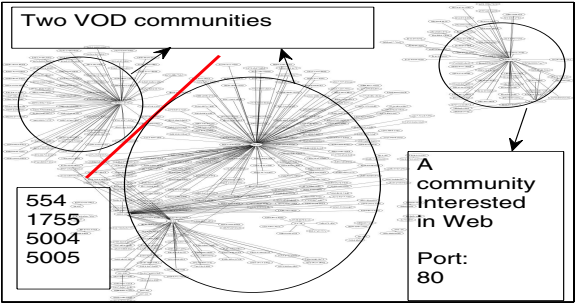


Fig. 5. Community interests example: VOD and Web

7 Demographics Study

Demographic study show current progress that CERNET2 has made towards IPv6 deployment. We demonstrate a brief demographic view in Figure 6. First, we demonstrate a geographic distribution of the IP addresses observed. Most IPv6 addresses observed in our study are from Asia and Europe. Second, we show traffic volume among CERNET2 and other IPv6 backbones. A thicker line indicates that there is more traffic. We can see that there is significant amount

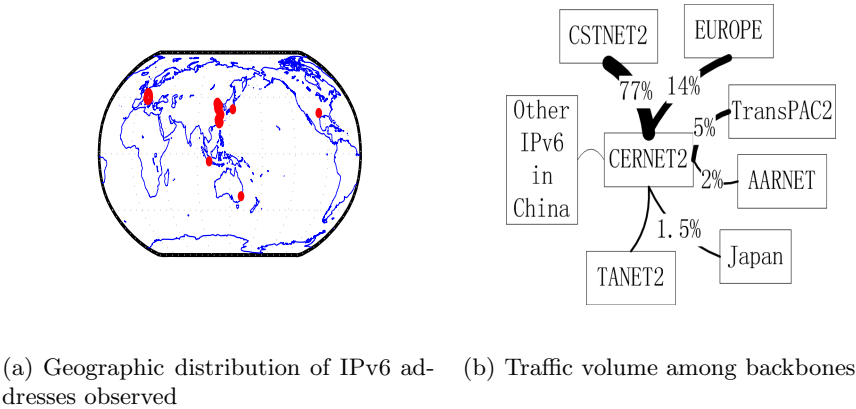


Fig. 6. Demographics CERNET2. Width of lines indicates traffic volume.

of traffic among CERNET2, CSTNET and Europe. Traffic between CERNET2 and CSTNET is mainly VOD and FTP traffic. Traffic between Europe and CERNET2 is mainly Web traffic. Main reason for users to use IPv6 networks is that there are services like Web, FTP and VOD. VOD servers of CERNET2 provide high speed low latency video programs and give users good experience. Promotions like free VoIP calls and FTP servers also give users alternative ways of making online calls and downloading resources.

8 Conclusion

We present our observations and conclusions of CERNET2 traffic by identifying host communities and interests. We analyze traffic by identifying flows and grouping hosts into communities using community finding algorithms. Result shows that the killer application is VOD and other types of services such as FTP and Web also generate a lot of traffic. By analyzing host participations, we find out that CERNET2 play role of producer and most heavy hitters are servers in CERNET2, which is also confirmed by our demographic study of traffic. Our study provides insight into future IPv6 deployment and network provision.

References

1. Li, L., Alderson, D., Willinger, W., Doyle, J.: A first-principles approach to understanding the internet's router-level topology. In: SIGCOMM 2004: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications, pp. 3–14. ACM Press, New York (2004)
2. Reittu, H., Norros, I.: On the power-law random graph model of massive data networks. *Perform. Eval.* 55(1-2), 3–23 (2004)
3. Crotti, M., Dusi, M., Gringoli, F., Salgarelli, L.: Traffic classification through simple statistical fingerprinting. *SIGCOMM Comput. Commun. Rev.* 37(1), 5–16 (2007)
4. Xu, K., Zhang, Z.L., Bhattacharyya, S.: Profiling internet backbone traffic: behavior models and applications. In: SIGCOMM 2005: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications, pp. 169–180. ACM Press, New York (2005)
5. Andrade, N., Mowbray, M., Lima, A., Wagner, G., Ripeanu, M.: Influences on cooperation in bittorrent communities. In: P2PECON 2005: Proceeding of the 2005 ACM SIGCOMM workshop on Economics of peer-to-peer systems, pp. 111–115. ACM, New York (2005)
6. Karagiannis, T., Papagiannaki, K., Faloutsos, M.: Blinc: multilevel traffic classification in the dark. In: SIGCOMM 2005: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications, pp. 229–240. ACM Press, New York (2005)
7. Bonfiglio, D., Mellia, P., Meo, M., Rossi, D., Tofanelli, P.: Revealing skype traffic: when randomness plays with you. In: Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications, pp. 37–48 (October 2007)

8. Yuan, L., Chuah, C.N., Mohapatra, P.: Progme: towards programmable network measurement. In: SIGCOMM 2007: Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications, pp. 97–108. ACM Press, New York (2007)
9. Chen, Y., Tsai, F.S., Chan, K.L.: Blog search and mining in the business domain. In: DDDM 2007: Proceedings of the 2007 international workshop on Domain driven data mining, pp. 55–60. ACM, New York (2007)
10. Qamra, A., Tseng, B., Chang, E.Y.: Mining blog stories using community-based and temporal clustering. In: CIKM 2006: Proceedings of the 15th ACM international conference on Information and knowledge management, pp. 58–67. ACM, New York (2006)
11. Gedeon, T., Parker, A.E., Campion, C., Aldworth, Z.: Annealing and the normalized n-cut. *Pattern Recogn.* 41(2), 592–606 (2008)
12. Scripps, J., Tan, P.N., Esfahanian, A.H.: Node roles and community structure in networks. In: WebKDD/SNA-KDD 2007: Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis, pp. 26–35. ACM Press, New York (2007)

DTPD: Data Transfer Tool Performance Diagnosis System in High Speed Networks*

Jong-Myoung Kim¹, Young-Ju Han¹, Giljae Lee²,
Woojin Seok², and Tai-Myoung Chung¹

¹ Internet Management Technology Laboratory,
Department of Electrical and Computer Engineering, Sungkyunkwan University,
300 Cheoncheon-dong, Jangan-gu, Suwon-si, Gyeonggi-do, 440-746, Republic of Korea
Tel.: +82-31-290-7222; Fax: +82-31-299-6673

{jmkim, yjhan, tmchung}@imtl.skku.ac.kr

² HPcN Project Div, HPcN Development Team,
Korea Institute of Science and Technology Information (KISTI),
335 Gwahangno, Yuseong-gu, Daejeon, Republic of Korea
Tel.: +82-42-869-0961; Fax: +82-2-3299-6067

{giljael, wjseok}@kisti.re.kr

Abstract. As the need for large data transmission is increased, the speed of network is also increased. Although the performance of the network is improved, the real data transmission over the high performance network is not improved as we expected. This is because most of host systems are optimized for the old low speed network. Indeed, the host system itself acts as the bottleneck. There are many researches to solve this problem. NDT and NPAD are the most famous projects. To achieve a high speed transmission, they provide a method for diagnosing problems in the network and end-systems that are the most common causes of all severe performance degradation over end-to-end paths. In spite of these efforts, the performance of data transmission is still limited if some data transfer tools such as SCP, SFTP, RSYNC are used. The reason is that the data transfer tools have some internal data transmission control mechanisms which can limit the transmission rate. In this case, end users have to use another efficient data transfer tool but it is difficult for them to find out the internal problem of the tools. Thus, experts have to provide a method for testing user's tools and guidelines on tool selection to the end users. In this paper, we propose the DTPD which diagnose the data transfer tool by monitoring the window size of the data transmission connection between two end systems. By using the DTPD, end users can easily test their tools through web pages and may choose more efficient data transport tools.

1 Introduction

Recently, the speed of network has been tremendously increased as the need for bulk data transmission is increased. Although the network infrastructure has grown greatly

* "This research was supported by the MKE(Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Advancement)" (IITA-2008-C1090-0801-0028).

in bandwidth, the end user cannot use the full advantage of the improved network. Indeed, the data transmission speed is limited, despite of the fact that the underlying network can support higher data transfer rate. This is because the configuration of the host system and its application are optimized for the low speed network.

Many researches have been done for dealing with this problem. NDT(Network Diagnostic Tool)[1] and NPAD(Network Path and Application Diagnosis)[2] are the representative projects. The purpose of these projects is the development of the network performance diagnosis tool to diagnose the network performance problems and to provide some solutions for efficient data transmission. To achieve a high speed transmission, they provide a method for diagnosing problems in the network and end-systems which are known for the most common causes of all severe performance degradation over end-to-end paths. By using these tools, end users can test their host and find the cause of low data transfer rate. The scope of diagnosis includes the fault TCP setting, wrong MTU size, duplex-mismatch, cable fault and so on. However, they only diagnose the problems under the transport layer of TCP/IP layer model so they cannot detect the problem that the data transmission tools may have. For example, SCP uses internal window mechanism just like TCP to control the overall data transmission and the window size of SCP is limited to 64Kbyte[11]. If the BDP(Bandwidth Delay Product) of the data transmission connection is larger than 64Kbyte, SCP usually acts as a bottleneck but general end users cannot know this fact. Experts have to advise them to use another efficient data transfer tool.

In this paper, we propose a DTPD(data transfer tool performance diagnosis system) that provides a method for deciding whether the data transfer tool is the bottleneck by monitoring the TCP window size of the data transmission connection. The end user can easily test their own transmission tool and may find out how their tool works efficiently by simply accessing the DTPD web page. Finally, DTPD will help the end users choose efficient data transfer tools.

The rest of this paper is organized as follows. In the next section, we review the related work on network performance diagnosis. In the section 3, we will introduce our system for diagnosing the data transfer tool performance. In section 4, we will verify our system by analyzing our test results. Finally, section 5 summarizes our paper and discusses about future research.

2 Related Works

This section divides into three parts. In the first part, the existing network performance diagnosis tools such as NDT and NPAD are summarized and their limitation is discussed. In the next part, the problem of data transfer tools are introduced from the viewpoint of performance. The last part introduces about web100 project[3] which is used in our DTPD.

2.1 Existing Network Diagnosis Tools

NDT. Network Diagnostic Tool[1] is one of the Internet2 projects to enhance the network performance. It is a server/client program that performs the end-to-end network

performance measurement and diagnosis based on the web100. The client is a java applet program and the end user can easily download the client program by accessing the NDT web page. Moreover, the interface type of the client program is GUI so the general end user can easily use function of NDT. By using NDT, the end users can find out what is wrong with their data transmission.

NDT tries to detect the cause of low data transfer at the end host or on the network. Currently, NDT only can perform the bottleneck detection and duplex mismatch detection but the WiFi detection and faulty hardware detection are planned to be included in the future[6].

NPAD. Network Path and Application Diagnosis[2] is a joint project of the PSC(Pittsburgh Supercomputing Center) and NCAR(National Center for Atmospheric Research). Similar to NDT, NPAD is a server/client program based on java applet. This project addresses the set of problems associated with end-hosts and their connections to a high speed backbone network. NPAD performs diagnosis about TCP tuning problem, duplex mismatch, cable fault, packet loss, and so on.

Limitation. Both NDT and NPAD can diagnose the problem only under the transport layer of TCP/IP model. If the data transfer tool acts as a bottleneck, NDT and NPAD cannot find out the cause of performance degradation. They even report a wrong diagnosis result to an end user who does not have enough knowledge. The wrong report may make the end user waste their time and labor.

2.2 Problem of Data Transfer Tools

FTP, SCP, SFTP and RSYNC are most widely used for data transfer. However, it is well known that SCP, SFTP and RSYNC have a problem with high speed transfer.

SCP and SFTP transport data by using SSH. For a data transmission over SSH, SSH creates a new channel over the TCP connection between source host and destination host. Then, the data is transferred over the channel. For reliable data transmission, each channel includes a flow control mechanism such as window mechanism just like TCP window mechanism. The internal window size of SSH is 64Kbyte and this window size can limit the data transmission rate if the BDP of the transmission connection is larger than the window size[11].

In the case of RSYNC[12], data can be transferred over SSH or RSYNC DAEMON. If RSYNC transfers data over SSH, the data transmission rate will be also limited by the internal window size of SSH.

Besides the window limitation of data transfer tools, there are many factors that affect the data transmission rate. The overhead of encryption and decryption can be the cause of bottleneck and compression and decompression can also make the transmission slow. Indeed, there is always a probability that the internal control mechanism of the data transfer tool can be the cause of the bottleneck.

2.3 Web100

Web100[3,8] is one of the NSF(National Science Foundation) projects to enhance the TCP performance that is optimized for the 10Mbps network by analyzing the TCP.

To achieve this purpose, web100 provides users with interfaces for collecting the TCP-related kernel variables for each TCP connection. Users can monitor the kernel variables of one TCP connection and can analyze the operation of TCP. Especially, web100 is used for automatic TCP buffer tuning and currently this work has been merged with the Linux kernel[5].

3 The Proposed System: DTPD

In this section, we propose the DTPD(data transfer tool performance diagnosis system) to overcome the limitation of the existing diagnosis tools. By using DTPD, end user can easily diagnose their own data transfer tool.

3.1 Diagnosis Method

As we discussed in section 2.2, data transfer tools can be the cause of the low transmission rate. However, the diagnosing the data transfer tool is very difficult, because the symptom(low data transmission rate) is identical with other problems like duplex mismatch, faulty TCP tuning, bottleneck and so on.

To diagnose correctly, we use X_Sndbuf and X_Rcvbuf that are the web100 variables. X_Sndbuf and X_Rcvbuf maintain the sender and receiver window size of a specific TCP connection and are limited by the predefined system variables, tcp_wmem and tcp_rmem which define the minimum, default and maximum size of the sender and receiver window. In general, these two web100 variables increase according to the BDP(Bandwidth Delay Product) that can be calculated following Equation (1). In other words, X_Sndbuf and X_Rcvbuf are increased within the range that is defined in tcp_wmem and tcp_rmem , if the capacity or RTT(Round Trip Time) is increased.

$$BDP = Capacity \times RTT \quad (1)$$

Although the BDP is increased, X_Sndbuf and X_Rcvbuf may not increased. If the transferred data from the application layer to the transport layer is not enough, the window size does not need to be increased, despite of the fact that the underlying network can support higher data transfer rate. This is a main idea of our diagnosis method. If the data transfer tool acts as the bottleneck, the data from the tool to transport layer will be not enough to increase the sender or receiver window size. Thus, we can diagnosis the problem of the data transfer tool by checking the X_Sndbuf and X_Rcvbuf are increased up to the maximum window size reasonably. If not increased, we can conclude that the tool has a problem that limits transmission rate.

However, there are two things we have to consider.

- If the BDP is smaller than the maximum window size, X_Sndbuf and X_Rcvbuf would not be increased up to the maximum size so we have to increase the BDP. To increase the BDP, we will use the NETEM[10] which can add delays to outgoing packets. If we delays the every outgoing packets, the RTT would be increased and finally we can get the sufficient BDP.

- If there is a congestion in the network, X_Sndbuf and X_Rcvbuf cannot be increased up to the maximum size. To consider this situation, we use the IPERF which is the end-to-end performance tester. The IPERF measures the transmission rate by sending dummy data without any internal control mechanism, so we can get the maximum sizes of X_Sndbuf and X_Rcvbuf under the current network situation. Thus, we can verify whether the data transfer tool is a bottleneck, by comparing the maximum X_Sndbuf and X_Rcvbuf of IPERF with the maximum X_Sndbuf and X_Rcvbuf of data transfer tool.

3.2 System Configuration

Figure 1 shows our system configuration. Like NDT and NPAD, end users can diagnose their data transfer tool through the java applet that can be downloaded by accessing the DTPD web page. For diagnosis, our system has to collect the web100 variables in the manager or the client. It would be a heavy overhead that the end user have to do the web100 kernel compile for collecting web100 variables. This deteriorate the useability and portability of our system. Thus, in our system, only the DTPD manager collects the web100 variables.

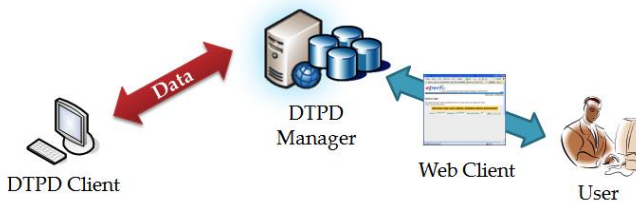


Fig. 1. System Configuration

Web Client: Web client is a java applet program to provide a GUI to end user. It can be downloaded by accessing DTPD web page. Web client simply forwards user's information that is necessary for testing and displays the results.

DTPD Manager: DTPD manager controls overall diagnosis. It controls the DTPD client for data transmission and collects the web100 variables: X_Sndbuf and X_Rcvbuf . It also performs the diagnosis process.

DTPD Client: Before testing, end user has to install the DTPD client in a user's host. The DTPD client is a daemon program that executes or stops user's data transfer tool under the request of the DTPD manager.

3.3 Operation

Figure 2 describes the operation of DTPD. The following list shows the description of each step in Figure 2.

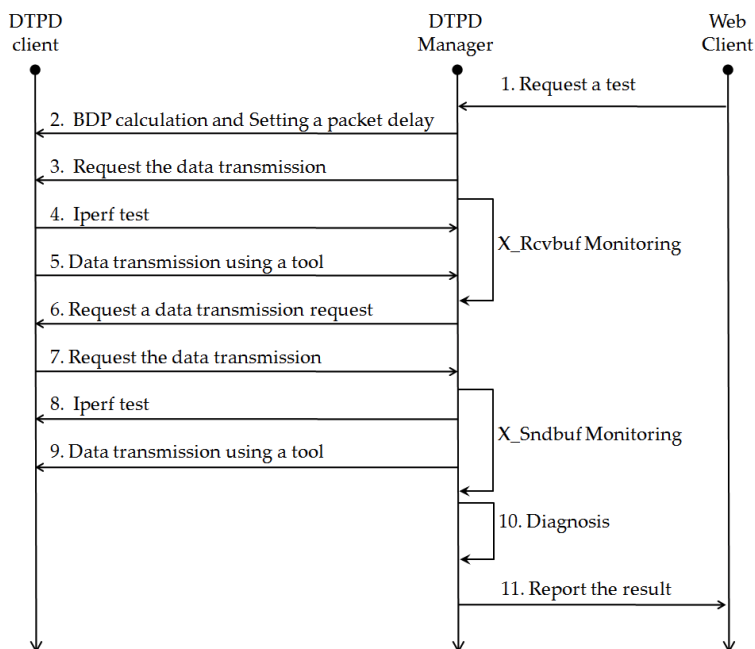


Fig. 2. DTPD Operation

1. End user requests the DTPD manager to diagnosis a specific data transfer tool α of a host A through the web client. The port number that will be used for diagnosis is also transferred.
2. The DTPD manager measures the RTT and path MTU to host A and calculates the BDP. If the BDP is smaller than the maximum window size which is defined in tcp_wmem and tcp_rmem , the DTPD manager increases the BDP by delaying the packets with NETEM.
3. The DTPD manager requests the host A to send the data to a specific port and starts to monitor the X_Rcvbuf .
4. The DTPD client starts the IPERF test.
5. The DTPD client starts the data transmission by using the tool α .
6. The DTPD manager stops the X_Rcvbuf monitoring and requests the data transmission request to pass the firewall which may exist at the host A .
7. The DTPD client requests the DTPD manager to start the data transmission. Once the DTPD manager receives this request, it starts to monitor the X_Sndbuf .
8. The DTPD manager starts the IPERF test.
9. The DTPD manager starts the data transmission by using the tool α .
10. The DTPD manager performs the diagnosis process based on the monitored web100 variables through the tool α and the IPERF.
11. The DTPD manager reports the diagnosis results to the user. If the tested tool has a bottleneck problem, some preferred data transfer tools are introduced.

4 Test

In this section, we verify our diagnosis method based on our test results. We tested our DTPD in GLORIAD(Global Ring Network for Advanced Application Development) which is a high-speed computer network used to connect scientific organizations in Russia, China, United States, the Netherlands, Korea and Canada. Figure 3 and Table 1 show our test environment. The capacity and RTT of the path between DTPD manager and client are 1Gbps and 0.2ms, so the BDP is about 25Kbyte which is smaller than the and maximum window size(4Mbyte). We monitored the *X_Sndbuf* and *X_Rcvbuf* while the data is transferred by using RSYNC, SCP, SFTP, FTP, IPERF. In the case of RSYNC, the data can be transferred with or without compression option over the SSH or RSYNC DAEMON, so the four cases are considered. To make the BDP lager than the maximum window size, we increase the RTT by usig the NETEM.



Fig. 3. Test Environment

Table 1. Window Size of DTPD Manger and Client, byte

Type	Min	Default	Max
<i>tcp_wmem</i>	4096	87380	4194304
<i>tcp_rmem</i>	4096	16384	4194304

Figure 4 and Table 2 show the test results when the RTT is 0.2ms, 10ms and 100ms. In all cases, the result of IPERF is best because it does not access the disk. When the RTT is 0.2ms, all tools have a good performance except RSYNC with compression option, and *X_Sndbuf* and *X_Rcvbuf* of the tools do not follow those of IPERF because the BDP is much smaller than the maximum window size. From this result, we can find out that the compression can be the cause of bottleneck but we cannot diagnose this problem by using *X_Sndbuf* and *X_Rcvbuf* because the BDP is not large enough to diagnose. As we explained in section 3.1, we can diagnose this problem by increasing BDP.

When the RTT is 10ms or 100ms, only the FTP and RSYNC without compression option over RSYNC DAEMON have a good performance. The performance of tools using the SSH is degraded due to the internal window size of the SSH. As we see in Figure 4(c), 4(d), 4(e) and 4(f), *X_Sndbuf* and *X_Rcvbuf* of the FTP and RSYNC without compression option over RSYNC DAEMON follow those of the IPERF. Indeed, the window size of the tools that have a good performance increases as the window size of IPERF increases. This results shows that our diagnosis method is valid.

The previous test was performed in the local network so the effect of cross-traffic is very small. Thus, we performed one more test to consider the cross traffic effect.

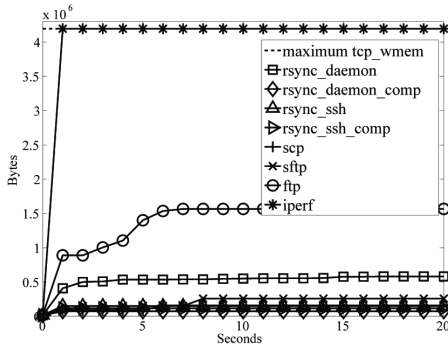
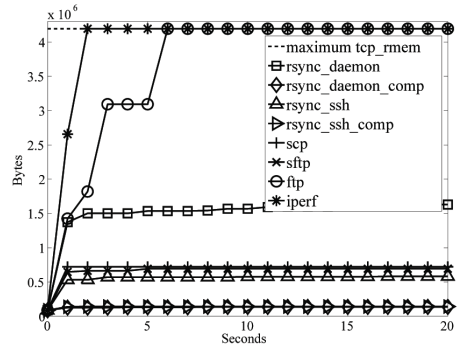
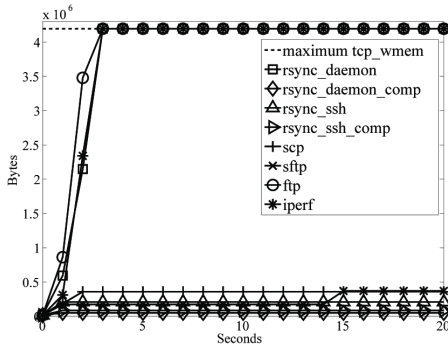
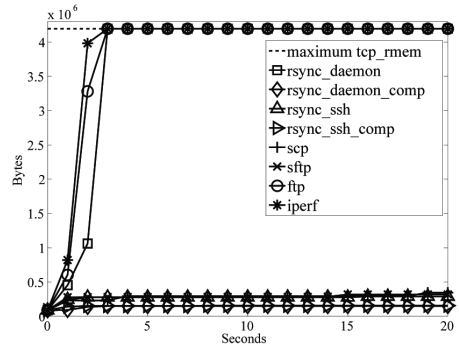
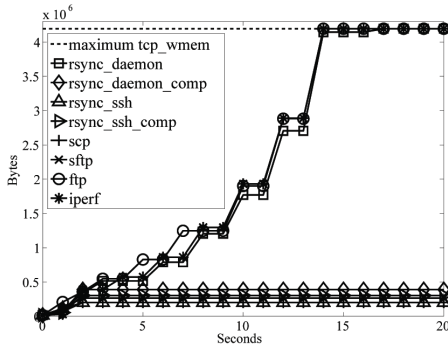
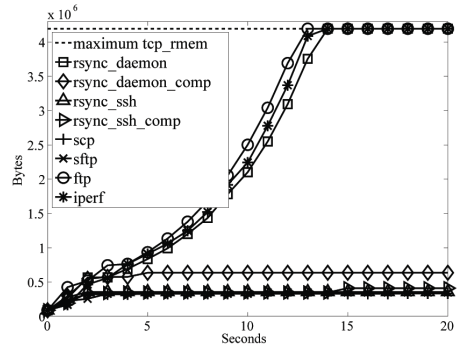

 (a) X_Sndbuf where RTT is 0.2ms

 (b) X_Rcvbuf where RTT is 0.2ms

 (c) X_Sndbuf where RTT is 10ms

 (d) X_Rcvbuf where RTT is 10ms

 (e) X_Sndbuf where RTT is 100ms

 (f) X_Rndbuf where RTT is 100ms

 Fig. 4. X_Sndbuf and X_Rcvbuf

We located one host on Daejeon in Korea and the other host on Seattle in USA. The capacity and RTT of the path is 1Gbps and 228ms. The results are shown in Figure 5 and Table 3. Similar to the results of 10ms and 100ms, X_Sndbuf and X_Rcvbuf of the FTP and RSYNC without compression option over RSYNC DAEMON, follow those of the IPERF. However, the RSYNC with compression option over RSYNC DAEMON

Table 2. Data Transmission Rate, Mbps

RTT	rsync_daemon	rsync_daemon_comp	rsync_ssh	rsync_ssh_comp	scp	sftp	ftp	iperf
0.2ms	304	92.8	176	90.4	180.8	272	353	936
10ms	296	92.8	44	90.4	48.8	50.4	485	928
100ms	184	92.8	5.4	41.6	6.4	6.2	172	235

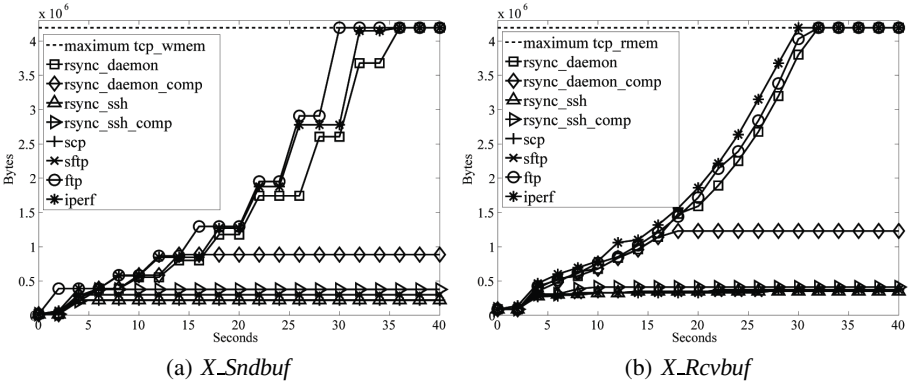


Fig. 5. Test Result between Daejeon and Seattle

Table 3. Data Transmission Rate, Mbps

Type	rsync_daemon	rsync_daemon_comp	rsync_ssh	rsync_ssh_comp	scp	sftp	ftp	iperf
Average	28.4	78.4	2.2	16.8	2.6	2.6	35.6	35.6
After 30s	97.3	79.2	2.2	17	2.6	2.6	98.7	104

is the fastest during 40 seconds even if *X_Sndbuf* and *X_Rcvbuf* of the RSYNC are not similar to those of the IPERF. There are two reasons for this phenomenon. The first reason is that the compression efficiency precedes the compression overhead when the data transmission of underlying network is slower than the speed of compression. The second reason is that the window is increased slowly when the RTT is large due to the TCP window mechanism. The performance of the IPERF, FTP and RSYNC without compression option over RSYNC DAEMON depends on the window size, so the transmission rate grows slowly. Actually, after 30 seconds, the IPERF, FTP and RSYNC without compression option over RSYNC DAEMON are faster than the RSYNC with compression option over RSYNC DAEMON as shown in Table 3. Based on these reasons, we can conclude that the compression of RSYNC can be the cause of bottleneck if the window size increases sufficiently. Thus, our diagnosis method can be effective if the enough transmission duration is guaranteed. To ensure this, the sufficient test duration must be guaranteed.

5 Conclusion

In this paper, we presented a data transfer tool performance diagnosis system that provides a method for deciding whether the data transfer tool is the cause of bottleneck by monitoring the TCP window size. We proved the effectiveness of our diagnosis method by analyzing our test results. The end user can easily test their tool and may find that the tool can be the bottleneck. Eventually, the end user can choose an efficient data transfer tool. Our study can be also applied to the work for developing a high speed data transfer tool.

A further direction of this study will be applying our DTPD to other various data transfer tools and finding the way to solve the problems of data transfer tools.

References

1. Web Page, <http://e2epi.internet2.edu/ndt/>
2. Web Page, <http://www.ucar.edu/npad/>
3. Web Page, <http://www.web100.org/>
4. Web Page, <http://www.psc.edu/networking/projects/applight/>
5. Ravier, C., Bennett, B.: High speed bulk data transfer using the SSH protocol. In: Proceedings of the 15th ACM Mardi Gras conference (2008)
6. Technical Paper, Network Diagnostic Tool (NDT): An Internet2 Cookbook, Internet2
7. Web Page, <http://dast.nlanr.net/Projects/iperf/>
8. Mathis, M., Heffner, J., Reddy, R.: Web100: Extended TCP Instrumentation for Research, Education and Diagnosis. ACM Computer Communications Review (July 2003)
9. Semke, J., Mahdavi, J., Mathis, M.: Automatic TCP Buffer Tuning. In: ACM SIGCOMM (October 1998)
10. Web Page, <http://www.linux-foundation.org/en/Net:Netem>
11. Ravier, C., Stevens, M.: Application Layer Network Window Management in the SSH Protocol. In: Supercomputing Conference (2004)
12. Web Page, <http://samba.anu.edu.au/rsync/>

QoS-Aware Customer Network Management (Q-CNM) System for Efficient Handovers with PMIPv6 and MIH

Young-Chul Jung and Young-Tak Kim*

Dept. of Info. & Comm. Eng., Graduate School, Yeungnam University.
214-1, Dae-Dong, Gyeongsan-si, Gyeongsanbuk-do, 712-749, Korea
callaguy@yumail.ac.kr, ytkim@yu.ac.kr

Abstract. Many researches for PMIPv6 have been focused on reducing handover latency and packet loss ratio in mobile network. The handover with QoS-aware differentiated service provisioning with load balancing, however, has not been studied in depth. In this paper, we propose a QoS-aware Customer Network Management (Q-CNM) system for QoS-aware handover with Proxy Mobile IPv6 (PMIPv6) and IEEE 802.21 Media Independent Handover (MIH). The Q-CNM supports the PMIPv6-based handover with QoS-aware load balancing for QoS-guaranteed network based handover in proxy mobile IP domain. Especially, the Q-CNM provides the MIH Information Service (MIIS) that collects network resource information for QoS-aware load balancing, QoS-guaranteed differentiated service provisioning, and centralized resource management in the proxy mobile IP domain. The proposed Q-CNM, PMIPv6 and MIH modules have been implemented on OPNET network simulator, and we analyze the QoS-aware handover performance based on the simulation results.

Keywords: PMIPv6, MIH, QoS, and Q-CNM.

1 Introduction

The Proxy Mobile IPv6 (PMIPv6) [1], that has been standardized by IETF Network-based Localized Mobility Management (NetLMM) working group, is receiving increased interests because it can support network-initiated handover without any involvement of a mobile node (MN) in the handover signaling. The proxy mobility agent in PMIPv6 performs the mobility management on behalf of the mobile node. There are a number of studies that try to provide fast handover for PMIPv6, mostly based on the performance improvement in the layer 3 handover sequence and the layer 2 handover signaling. The handover with QoS-aware load balancing, however, has not been studied in depth.

Additional performance enhancements in PMIPv6-based handover have been proposed using the IEEE 802.21 Media Independent Handover (MIH) [2] to provide event triggering service, command service for management and control of link behavior, and media independent information service for effective system access and effective handover decisions. The detailed operation of the MIH Information Service (MIIS) has not been studied in depth.

* Corresponding author.

In this paper, we propose a QoS-aware Customer Network Management (Q-CNM) system for QoS-aware handover with Proxy Mobile IPv6 (PMIPv6) and IEEE 802.21 Media Independent Handover (MIH). The Q-CNM supports the PMIPv6-based handover with QoS-aware load balancing for QoS-guaranteed network based handover in proxy mobile IP domain. Especially, the Q-CNM provides the MIH Information Service (MIIS) that collects network resource information for QoS-aware load balancing, and centralized resource management in the proxy mobile IP domain. The proposed Q-CNM, PMIPv6 and MIH modules have been implemented on OPNET network simulator, and we provide the performance analysis based on the simulations.

The rest of this paper is organized as follows. Section 2 briefly explains the related work on PMIPv6 and MIH. Section 3 explains the proposed Q-CNM, including for the functional architecture, interaction scenario, and message exchanges among the major PMIPv6 nodes and Q-CNM. Section 4 evaluates the performance of the QoS-aware handover with PMIPv6, MIH and Q-CNM. Finally, section 5 concludes this paper.

2 Related Work

2.1 Proxy Mobile IPv6 (PMIPv6)

The PMIPv6[1, 3] that has been standardized by IETF NETLMM provides several advantages compared to FMIPv6 and HMIPv6. One of the most attractive features of PMIPv6 is its network-based handover supporting without involvement of the mobile node (MN) in the signaling required for handover process. The Mobility Access Gateway (MAG) in the network performs the handover-related signaling and does the mobility management on behalf of the MN [2]. The Local Mobility Anchor (LMA) is a modified MIPv6 Home Agent (HA), which maintains the address bindings for MNs in PMIPv6 domain.

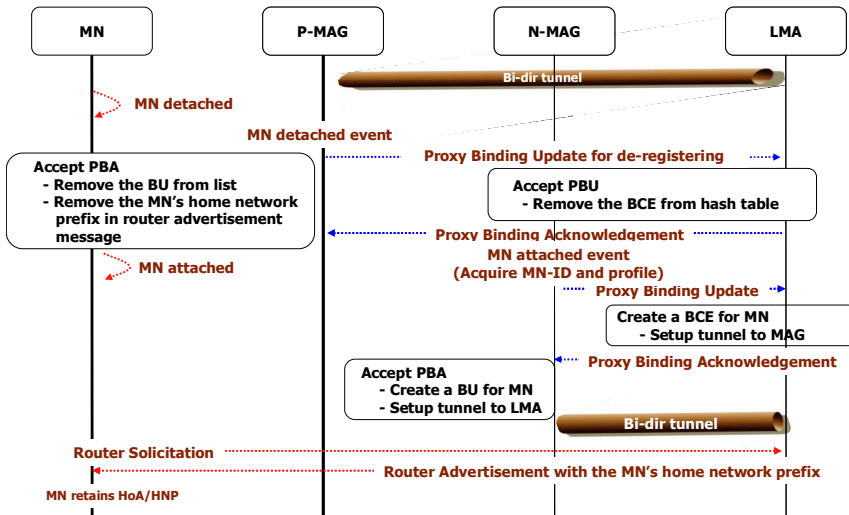


Fig. 1. The handover procedure in PMIPv6

Because of this MN-independent mobility support, the PMIPv6 can be used for the legacy node with IPv6 protocol stacks without MIPv6 functions. Also, the network-initiated handover can provide better network performance since it can make handover decision for each mobile node based on the current traffic load at each access point (AP) and switch/router on the expected path from LMA. Fig. 1 shows the handover procedure defined by [1].

2.2 Media Independent Handover (MIH) Services

IEEE 802.21 Media Independent Handover (MIH) [2] defines supporting functions to improve mobility between heterogeneous networks. MIH provides link layer intelligence and other related network information to upper layers. The Media Independent Handover Function (MIHF) provides abstracted services to higher layers by means of a unified interface. This unified interface exposes service primitives that are independent of the access technology. The MIHF defines three main asynchronous and synchronous services: the *Media Independent Event Services* (MIES), the *Media Independent Command Services* (MICS), and the *Media Independent Information Services* (MIIS).

In order to support QoS-aware handover with PMIPv6 and MIH, the MIIS should be able to collect the network resources in the PMIPv6 domain, including the current traffic load at each MAG and APs. In this paper, we propose the system architecture that provides interworking functionality between Q-CNM and MIIS Server.

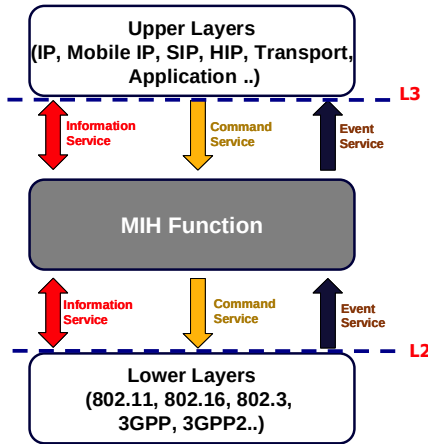


Fig. 2. MIH functional architecture

3 QoS-Aware Customer Network Management (Q-CNM) System for PMIPv6 with MIH

3.1 Proposed Configuration for QoS-Aware Handover Provisioning with PMIPv6 and MIH

Fig. 3 depicts an example of PMIPv6 domain with IEEE 802.11e WLAN for QoS-aware DiffServ provisioning. In this network configuration, the LMA and MAG are

connected by MPLS TE-LSP that can provide bandwidth-guaranteed tunneling. IEEE 802.1q VLAN (virtual LAN) is configured between the MAG and APs. For QoS-aware differentiated service provisioning in IEEE 802.11 WLAN basic service set (BSS), we assume the AP supports IEEE 802.11e EDCA (Enhanced DCF Channel Access) that provides differentiated distributed access to the wireless medium for 8 priorities of stations [4, 5]. EDCA defines the access category (AC) mechanism that supports the differentiated priorities at the stations. Each station may have up to 4 ACs (AC_VO, AC_VI, AC_BE and AC_BK) to support 8 user priorities. One or more user priorities are assigned to one AC.

In Fig. 3, Q-CNM provides the intranet network management for QoS-aware differentiated service provisioning. The details of Q-CNM will be provided in section 3. The QoS-aware SIP (session initiation protocol) / SDP (session description protocol) (Q-SIP/SDP) [4] proxy server receives the QoS session establishment request message from the MN stations, and informs the QoS session information to the Q-CNM for resource management and QoS-aware handover.

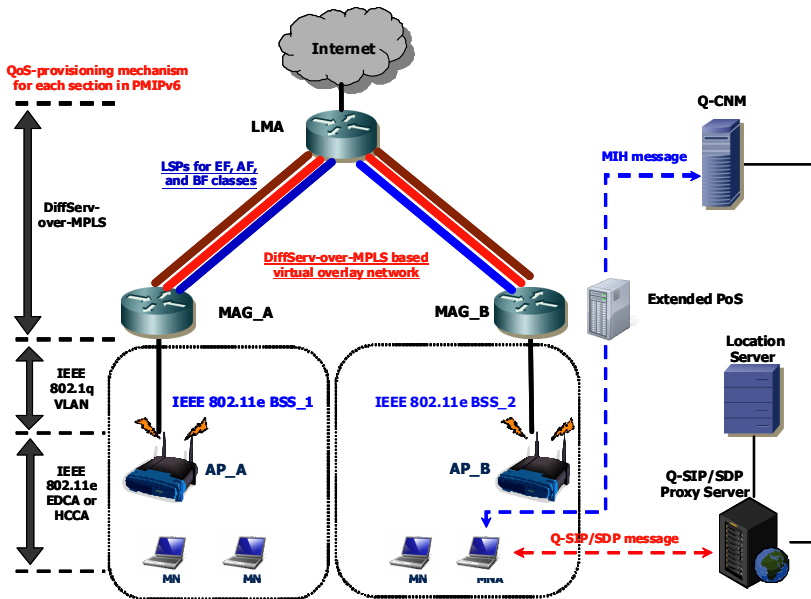


Fig. 3. Configuration for QoS-aware handover provisioning with PMIPv6 and MIH

3.2 Functional Architecture of Q-CNM System for PMIPv6 with MIH

Fig. 4 depicts the functional architecture and interactions of Q-CNM. Q-CNM contains functional blocks of QoS-aware session manager, service policy, PMIPv6 manager, home point of service (PoS), MIIS server, intranet management (i.e., resource/connection, performance, fault, configuration, and event indication provider), command line interface (CLI) providers for subnetworks, SNMP manager and SNMP trap handler for event reporting. The resource/connection manager firstly discovers and collects the network resource information in the PMIPv6 domain network. It

collects information on the total capacity of each link among PMIPv6 nodes (such as LMA, MAGs, and APs), the network topology, current traffic load & established QoS-aware sessions at each node, and available bandwidth. It also continuously manages the allocated and available resources at each node, and provides the network resource information to the MAG through MIIS that performs the QoS-aware network-based handover decision making for mobile nodes.

The QoS-aware session manager receives QoS-aware session establishment request from the Q-SIP/SDP [6]. Since the IEEE 802.11e AP is not supporting connection-oriented session establishment, the EDCA channel allocation for each QoS-aware session establishment must be handled by Q-CNM via IEEE 802.11e CLI provider. It also provides connection admission control (CAC) function for QoS-guaranteed session establishment at each QoS-aware BSS (Q-BSS) by a IEEE 802.11e AP.

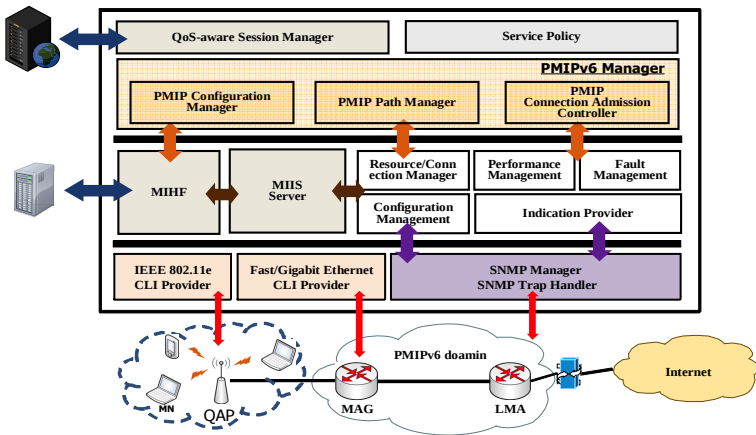


Fig. 4. PMIPv6 with Q-CNM

The PMIPv6 manager is logically located above intranet-manager and has three main modules, namely PMIP configuration manager, PMIP path manger, and PMIP CAC. PMIP configuration manger performs to collect configuration information on PMIP network entity, such as LMA, MAG, and AP. Based on collected configuration information, PMIP path manager configures and maintains bi-directional QoS-paths from APs to LMA. The QoS-path contains the information of each network area, which composes of entire QoS-path, through the intranet manager. PMIP CAC carries out checking of available network resources for the requested QoS-path. The MIIS server in the proposed scheme collects dynamic information such as current network load, and static information such as IP assignment method as well, through interworking with intranet manager.

Fig. 5 and Fig. 6 depict the entire procedure to support QoS-guaranteed handover in the proposed system. When the MN is located in *link_going_down* condition, it sends the event to Extended PoS (EPoS) by using MIH protocol. EPoS then requests scanning action to MN and gathers the list of available APs. Based on the list of available APs, the EPoS requests available resource checking for each candidate AP to Q-CNM. At this point, Q-CNM finds the proper QoS-path based on the candidate AP

and checks the QoS capability for selected QoS-path. After receiving CAC result from Q-CNM, the EPoS selects target AP that has the highest available network resource. Finally, the EPoS requests the mobile node to perform handover to the selected target AP.

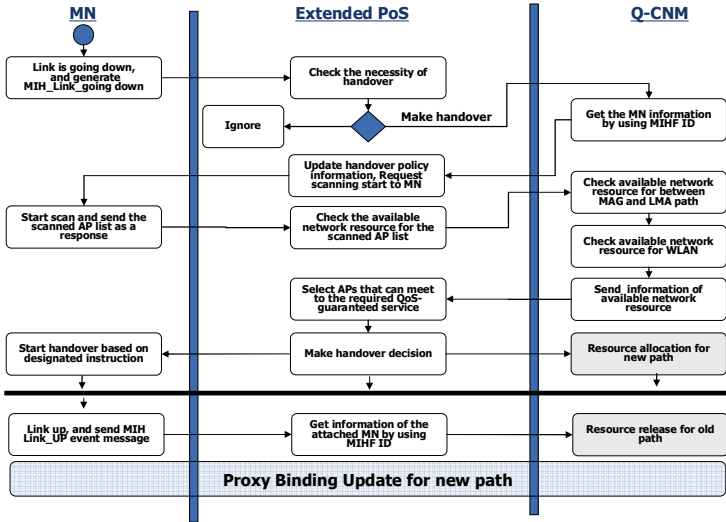


Fig. 5. Network resource allocation by Q-CNM for PMIPv6 with MIH

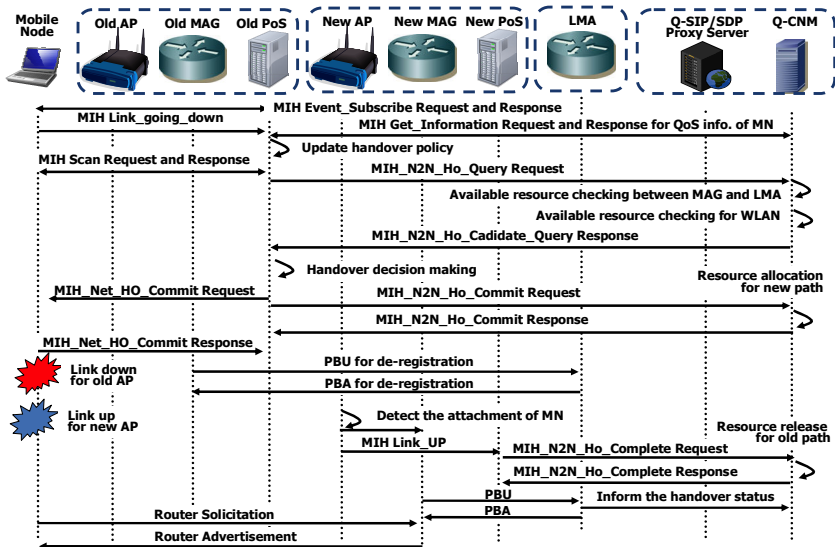


Fig. 6. QoS-aware handover with PMIPv6, MIH and Q-CNM

3.3 QoS-Guaranteed Handover Decision Maker(QHDM)

Fig. 7 and 8 show algorithms of the proposed QoS-guaranteed Handover Decision Maker (QHDM). QHDM is targeted to support QoS-aware handover decision making and to provide load balancing among WLAN subnetworks. In order to perform QoS-aware handover decision making, we collect QoS information, such as required bandwidth of mobile node and current available resources in the selected QoS-path, through the proposed Q-CNM with Extended PoS. Based on the collected QoS information, QHDM selects the target AP considering the load balancing among APs.

QHDM is divided into two functions, namely *QoS-aware HandoverDecisionMaking* and *CACatQCNM*. The *QoS-aware HandoverDecisionMaking* function located at EPoS carries out collecting information for candidate APs and deciding the target AP, through interworking with Q-CNM. *CACatQCNM* located at Q-CNM selects the QoS capable path (from AP to LMA) for each candidate AP and performs CAC function for each network section that composes the entire QoS-path.

```

Procedure QoS-aware HandoverDecisionMaking(MN_MIHF_ID)
{
1  // Request scanning to the MN by using MIH protocol
   ScannedAPs []  $\leftarrow$  ScanbyMN(MN_MIHF_ID)
2  FOR i=0 TO Numscanned_ap DO
3    IF ScannedAPs[i].RSSI  $\geq$  RSSImin_threshold THEN
4      CandidateAPs []  $\leftarrow$  ScannedAPs[i]
5    END IF
6  END FOR

7  // Check available bandwidth from AP to LMA
   QoSCapableAPs []  $\leftarrow$  CACatQCNM(MN_MIHF_ID, CandidateAPs [])
8  TargetAP  $\leftarrow$  MAX_AVAILABLE_BANDWIDTH(QoSCapableAPs [])
}
END Procedure

```

Fig. 7. Handover Decision Making Pseudocode in Extended PoS

Fig.7 shows the pseudocode of the *QoS-aware HandoverDecisionMaking* function. The first step of the algorithm is getting the list of available APs that can communicate with the current mobile node. In order to get candidate AP list, the function requests the mobile node to scan via MIH command service. The QHDM then decides candidate APs that has the RSSI value higher than the minimum RSSI threshold value. After making the set of candidate APs, QoS-Path selection and CAC provided by Q-CNM are performed for each candidate AP. Based on retrieved result from CAC at Q-CNM, finally, the QHDM selects target AP that has the highest available network resource.

The pseudocode in Fig. 8 shows the algorithm for *CACatQCNM* function. When the function is invoked by *QoS-aware HandoverDecisionMaking* function, as first step, it acquires the QoS information of the current mobile node from PMIP path manager. The QoS-path information, that contains network resource information from

```

Procedure CACatQCNM(MN_MIHF_ID, CandidateAPs [])
{
1 // Get required bandwidth info. of the MN that is stored in PMIP path manager
  RequiredBandwidth  $\leftarrow$  QoS[MN_MIHF_ID].bandwidth
2 FOR j=0 TO Numcandidate_ap DO
3   // QoSPathInfo contains the available bandwidth of each network area
   // that compose of the entire QoS path
   QoSPathInfo  $\leftarrow$  QoSPathToLMA(CandidateAPs[j])
4   IF RequiredBandwidth is available in QoSPathInfo THEN
5     QoSCapableAPs []  $\leftarrow$  CandidateAPs[j]
6   END IF
7 END FOR
8 // Sort order by available bandwidth of AP to provide load balancing in WLAN
  Sort(QoSCapableAPs [])
9 RETURN QoSCapableAPs []
}
END Procedure

```

Fig. 8. QoS-path Selection and CAC Pseudocode in Q-CNM

AP to LMA, is collected for each candidate AP. At this point, the CAC function is performed for each network area in the entire QoS-path. After completing CAC functions, the function sorts the candidate APs ordered by available bandwidth of AP to provide load balancing in WLAN. Finally, the list of candidate APs that fulfills the required QoS is delivered to the QHDM.

4 Performance Evaluation and Analysis

In this section, we analyze the proposed scheme using simulation modules implemented on OPNET [7] network simulator. In order to evaluate the proposed scheme,

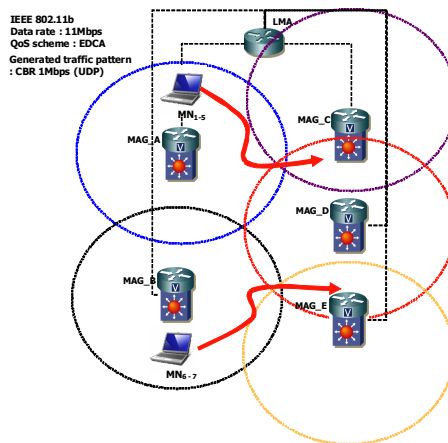


Fig. 9. Experiment Scenario

two kinds of simulation scenario are used. Both of scenarios have the same network topology but the first one is only configured with basic PMIPv6 system and the second scenario is configured with proposed scheme, which is composed of Q-CNM at PMIPv6 with MIH.

Fig. 9 depicts the experiment scenario. In this experiment, IEEE 802.11b/e (11Mbps/EDCA only) is used as wireless LAN system. We distribute 5 MAGs to the PMIPv6 domain and one AP is only attached to one MAG. The service distance of each AP is set to 100m and each cell has overlapped with other cells in order to provide service continuity. We used 7 MN and each MN generates 1Mbps CBR traffic by using UDP. All MNs are moved following by the trajectory shown in Fig. 9.

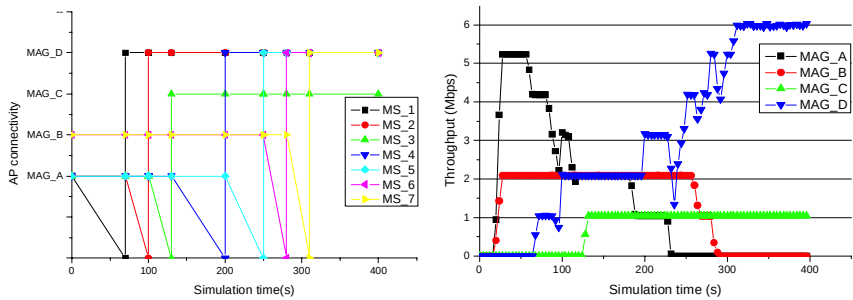


Fig. 10. AP connectivity and Throughput on Basic PMIPv6 System

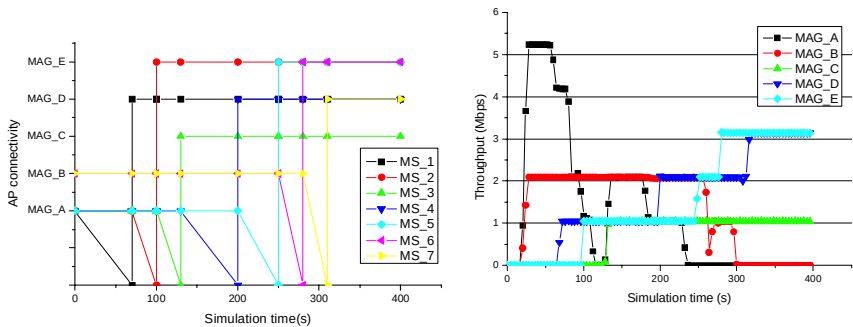


Fig. 11. AP connectivity and Throughput on Proposed System

Table 1. End-to-End Delay, Jitter, and Packet Drop Ratio at MN

	End-to-End Delay(ms)		End-to-End Jitter(ms)	Packet Drop Ratio
Basic PMIPv6	MAX	885	228	5%
	MIN	35	21	
	AVG	119	96	
Proposed Scheme	MAX	45	50	0.9%
	MIN	1.4	0.7	
	AVG	13	3	

Fig. 10 shows the simulation result for AP connectivity of each MN and throughput of each MAG with basic PMIPv6 mode. In this scenario, only the basic PMIPv6 system is used to provide mobility function to MN. In this case the QoS management system is not provided and MN may attach to the specific AP without considering the current available network resource because MN selects the target AP that has the highest beacon signal strength [8]. Consequently, connections can be established to a nearest AP without any load balancing, and network congestion is occurred in the AP. In Fig. 10, All MNs except MN 3 are attached to MAG_D, therefore MAG_D is congested, while other MAGs are under loaded.

Fig. 11 shows the AP connectivity and throughput of each MAG when the proposed Q-CNM with MIH is adopted to PMIPv6 system. By comparing with Fig. 10, when the MN makes handover at layer 2, the proposed QHDM in the EPoS checks the available bandwidth through Q-CNM and notifies the selected AP to perform QoS-guaranteed handover to the selected AP. Consequently, we can assure that all MNs are uniformly distributed to the available APs considering with current network resources.

Table 1 shows the performance analysis result in terms of delay, jitter, and packet loss ratio. All of performance parameters were measured at IP layer. Delay and jitter are significantly decreased (Avg. delay was 1.4ms and jitter was 0.7ms). In the case of packet loss ratio, it does not show the difference according to the characteristics of WLAN, such as retransmission and saturation of packet transmission.

5 Conclusion

In this paper, we proposed a QoS-aware-Customer Network Management (Q-CNM) system for QoS-aware handover in Proxy Mobile IPv6 (PMIPv6) with IEEE 802.21 Media Independent Handover (MIH). The Q-CNM supports the PMIPv6-based handover with QoS-aware load balancing for QoS-guaranteed network initiated handover in proxy mobile IP domain. Especially, the Q-CNM provides the MIH Information Service (MIS) that collects network resource information for QoS-aware load balancing, and centralized resource management in the proxy mobile IP domain. The proposed Q-CNM, PMIPv6 and MIH modules have been implemented on OPNET network simulator, and we provided the performance analysis based on the simulations. From the simulation results, we verified that our proposed system supports QoS-aware handover provisioning considering with traffic load balancing.

References

1. Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., Patil, B.: Proxy Mobile Ipv6, draft-ietf-netlmm-proxymip6-17.txt (May 27, 2008)
2. IEEE P802.21/D10.0 Draft standard for Local and Metropolitan Area Networks: Media Independent Handover Services (April 2008)
3. Lee, J.-H., Kim, Y.-D., Chung, T.-M.: An Investigation of Distance-Based Proxy Mobile Client Selection Schemes for NetLMM. In: AINAW 2007 (2007)

4. IEEE Std 802.11e, Part 11: Wireless LAN Medium Access Control (MAC) and Physical layer (PHY) specifications: Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements (2005)
5. Jung, Y.-C., Kim, B.-K., Kim, Y.-T.: Home/Office Intranet Resource Management for QoS-guaranteed Realtime Stream Service Provisioning on IEEE 802.11e WLAN. In: Proceedings of NOMS 2008, Salvador, Brazil, 09 April 2008, pp. 959–962 (2008)
6. Kim, Y.-T., Jung, Y.-C., Kim, S.-W.: QoS-Guaranteed Realtime Multimedia Service Provisioning on Broadband Convergence Network (BcN) with IEEE 802.11eWireless LAN and Fast/Gigabit Ethernet. JCN 9(4) (December 2007)
7. <http://www.opnet.com>
8. Kim, B.K., Jung, Y.C., Kim, I., Kim, Y.T.: Enhanced FMIPv4 Horizontal Handover with Minimized Channel Scanning Time based on Media Independent Handover (MIH). In: BcN 2008 (April 2008)

Dynamic Multi-stream Transport Protocol*

Seung-Joon Seok¹, Hyeong-Jun Kim², Kwang-Min Jung³,
Kyung-Hoe Kim⁴, and Chul-Hee Kang⁴

¹ Dep. of Computer Engineering, Kyungnam University, Masan-si, Korea
sjseok@kyungnam.ac.kr

² LG Electronics, Seoul, Korea
hj286kim@lg.com

³ Samsung Electronics, Suwon-si, Korea
mini0912@msn.com

⁴ Dep. of Electronics and Computer Engineering, Seoul, Korea
{kyunghoe, chkang}@widecomm.korea.ac.kr

Abstract. Legacy TCP is not suitable in high speed network environments due to its slow response time to high speed link bandwidth (large scale window size). A number of solutions have been proposed to support scalability under high link bandwidth. However, other criteria such as fairness and friendliness, in addition to scalability, should be considered. Accordingly, this paper describes a transport protocol for high speed networks, referred to as DMS-TCP, which uses multiple end-to-end streams simultaneously and considers scalability, fairness and friendliness simultaneously. DMS-TCP describes a scheme that controls the number of streams according to network condition. In this paper, the performance of DMS-TCP is evaluated through NS-2 simulation. The simulation results demonstrate that DMS-TCP can be used efficiently in high speed networks.

Keywords: Multi-Stream, Transport Protocol, Dynamic, High-Speed Network.

1 Introduction

Legacy TCP deployments can not obtain high throughput in high speed networks due to the slow packet loss recovery times and need for low supporting loss rates. For example, in standard TCP, (TCP Reno) connection with 1500byte packets and a 100ms RTT, achieving a steady-state throughput of 10 Gbps, would require an average congestion window of 83,333 segments and a packet drop rate of at most one congestion event every 5,000,000,000 packets. The average packet drop rate of at most $2 \cdot 10^{-10}$, is needed for full link utilization in this environment, corresponding to a bit error rate of at most $2 \cdot 10^{-14}$, this is an unrealistic requirement for current networks[1]. It means that standard TCP can not obtain 10 Gbps in end-to-end throughput. Accordingly, TCP Reno may become a performance bottleneck in this environment. This is defined as a scalability problem. A number of protocols are proposed to address this problem. Usually, these

* This work was supported by 2008 research and test program using KOREN/APP/TEIN of National Information Society Agency (NIA).

protocols are interested only in obtaining large window size. However, Fairness and friendliness are important parameters to assess the proposed protocols, because a flow should share network resources with other flows. Fairness means that each connection receives an equal share of bandwidth, if more than one connection sharing a bottleneck link exists. Friendliness means that in TCP, long-term average throughput is not affected by a low bandwidth link.

This paper proposes a transport protocol to support high speed networks considering scalability, fairness and friendliness. This protocol is called DMS-TCP (Dynamic Multi-Stream TCP). DMS-TCP controls the number of streams in accordance with the available network bandwidth. Accordingly, the concept of DMS-TCP means that the multiple-stream can increase end-to-end throughput faster than standard single-stream TCP. In the other words, other protocols proposed for high speed networks allow a single-stream's congestion window to increase rapidly and obtain large delay-bandwidth product.

This paper is organized as follows: Section 2 describes previous work. Section 3 then describes the proposed protocol, DMS-TCP. Section 4 illustrates the characteristics of DMS-TCP and Section 5 evaluates performance of DMS-TCP by NS-2 simulation. Finally, this paper is concluded in Section 6.

2 Related Works on TCP Modification for High-speed Networks

Standard TCP employs a loss-based approach. It uses *AIMD* (additive increase multiplicative decrease), where the congestion window size is increased by one packet every RTT, and reduced to half itself on each packet loss event. Thus, bandwidth utilization is too low for high speeds because of increasing slowly and decreasing rapidly. Several protocols, such as High-Speed TCP [1, 2, 3], Scalable TCP [5], XCP [6] and FAST TCP [7], are proposed to treat the standard TCP deficiencies when operating in high-speed networks.

Scalable TCP (STCP) uses congestion control of *MIMD* (Multiplicative Increase Multiplicative Decrease) instead of *AIMD*. When the congestion window is larger than 16, STCP uses a fixed response parameter, such as in Eq. (1), where increase factor a is 0.01 and in Eq. (2), where decrease factor b is 0.125. In all other case, it operates in the same way as standard TCP.

$$cwnd \leftarrow cwnd + 0.01 \quad (1)$$

$$cwnd \leftarrow cwnd - (0.125 \times cwnd) \quad (2)$$

HSTCP uses a modified *AIMD*, which calculated from current window size. Eq. (3) is the increment value, which is the amount of congestion window size increased after an ACK arrives. Eq. (4) is the decrement value, which is the amount of congestion window size decrease after packet loss is detected.

$$a(w) = \frac{W^2 \cdot p(w) \cdot 2 \cdot b(w)}{2 - b(w)} \quad (3)$$

$$b(w) = (B - 0.5) \cdot \frac{\log w - \log W_0}{\log W_1 - \log W_0} + 0.5 \quad (4)$$

where w is the current window size. In HSTCP, the current window size is larger, $a(w)$ is larger and $b(w)$ is smaller.

XCP (eXplicit Control Protocol) provides efficiency, equity and stability, since it notifies congestion degree explicitly with procedure similar to the ER (Explicit Rate) algorithm of ABR (Available Bit Rate) in the ATM network. It generalizes the ECN (Explicit Congestion Notification) proposals. Routers monitor input traffic rate at each of their output queues. Based on the difference between link bandwidth and its input traffic rate, the router notifies the flows sharing the link to increase or decrease congestion windows. Consequently, the packet will contain feedback from the bottle-neck along the path. From these procedures, the router can control directly the congestion windows of the sources. XCP insists it has both an efficiency controller (EC) and a fairness controller (FC). However, there are some problems in deploying XCP as a high BDP (Bandwidth Delay Production) protocol. To run XCP in networks, XCP should be applied to the sender side, receiver side, and router at the same time.

FAST TCP uses congestion control based on Vegas [11]. FAST TCP uses queuing delay as its main measure of congestion in its window adjustment algorithm. FAST TCP uses the delay based congestion avoidance mechanism that analogizes the buffer state of networks with the momentary change of RTT, such as TCP Vegas. Under normal network conditions, FAST TCP periodically updates the congestion window W , based on the average RTT, according to Eq. (5)

$$W \leftarrow \min \left[2W, (1 - \gamma)W + \gamma \left(\frac{baseRTT}{RTT} W + \alpha \right) \right] \quad (5)$$

Where γ is a constant between 0 and 1, RTT is the current average RTT, $baseRTT$ is the minimum RTT observed so far, and α is the protocol parameter that controls fairness and the number of packets of each flow buffered in the network. FAST TCP adjusts its window by a large amount, up or down, when the number of buffered packets is far away from its target, and a small amount when it is close to its target.

Except XCP (a router assisted protocol), these protocols adaptively adjust their increase rates based on the current window size. The larger the congestion window is, the faster the growth is. These protocols are claimed to be TCP friendly, under high loss rate environments as well as highly scalable under low loss environments. These protocols mainly address the method of improving scalability. However, Fairness and friendliness are also important parameters to assess the proposed protocols, because a flow should share network resources with other flows using other protocols.

3 Proposed Scheme

This paper proposes the scheme using a multi-stream, through a single end-to-end path, which controls the number of streams for providing scalability. Thus, this is called Dynamic Multi-stream TCP (DMS-TCP). DMS-TCP also considers fairness and friendliness as well as scalability.

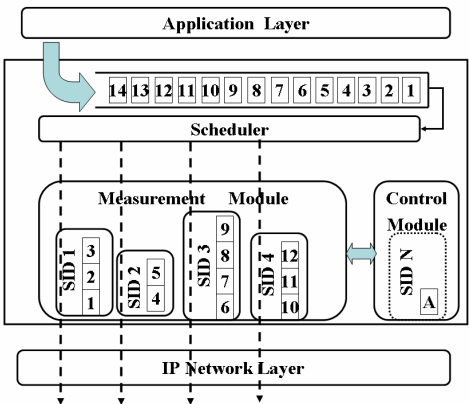


Fig. 1. DMS-TCP architecture

3.1 Dynamic Multi-Stream TCP (DMS-TCP)

DMS-TCP consists of three modules, as shown in Fig. 1: Scheduler, Measurement, and Control. The scheduler (weighted round robin) distributes packets to reduce the probability of out of order sequences. The weight of the scheduler is the window size of each stream. The control module determines the number of streams according to the information from measurement module and opens/closes streams. Each stream performs separately its congestion control. Thus, Stream IDentification (SID) and Stream Sequence Number (SSN) are added to the header of packet in order to prevent Head of Line (HOL) blocking.

In Fig. 2, there are two periods that are repeatedly operated in DMS-TCP: Measurement (T_m) and Control (T_c) periods. For each fixed measurement period, DMS-TCP's measurement module detects the end-to-end path conditions. Then the DMS-TCP starts a control period operation that opens/closes streams. After completing these operations, DMS-TCP enters into measurement periods again.

The measurement module collects information on the number of packet losses, and timeout, and decides the Overload Point (OP) is reached, based on the information. Here, the OP is reached when the current number of streams is sufficiently high that the path is congested. In this paper, if the number of packet losses is the same as the current number of streams or if timeout occurs in any streams, the OP point is reached. DMS-TCP will then decrease the number of streams proportionally, based on the current number of streams. This measured information should be renewed after every measurement period.

The mechanism of the control module is depicted in Fig. 2. The number of streams can be increased or decreased according to four modes: Base, Start-Up, Balance and Overload modes. DMS-TCP starts from base mode. In this mode, DMS-TCP starts packet transfer through a single-stream, in the same way as standard TCP. In start-up mode, the number of streams is increased exponentially until its Start-Up Stream Threshold (Th_{sus}). Then, it goes into balance mode; and the number of streams is incremented. Finally, the number of streams is decreased by the Decrease Rate (DR),

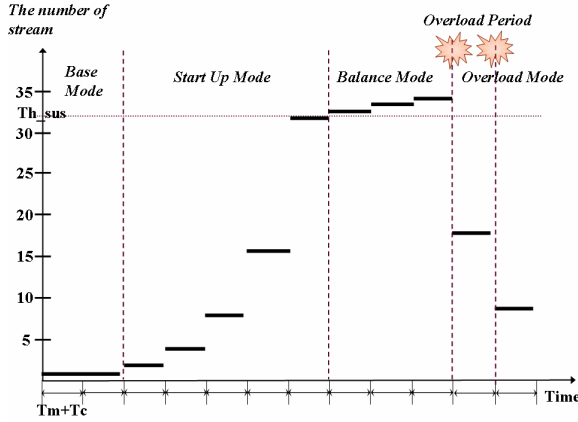


Fig. 2. Concept of DMS-TCP

in the overload mode. To open or close streams, the sender's control module sends 'add-stream' or 'del-stream' segments to the receiver, respectively. In the next subsection, we explain these modes in detail.

3.2 Control Modes of DMS-TCP

There are four modes available to tune the number of streams. Each mode is selected through measured information by a control module (Fig. 3). These four modes are detailed below.

Base Mode. This mode maintains a single-stream. When the current mode is in base mode, it means that the network is not operating at high speed or DMS-TCP is just started. This is important in providing friendliness to standard TCP. At this time DMS-TCP acts as standard TCP.

Start-Up Mode. The control module conducts start-up mode operation, if the current number of streams is below Th_{sus} and the OP is not detected after the measurement period. In start-up mode, the number of streams is doubled every control period. The Th_{sus} parameter denotes the maximum number of streams in start-up mode. This mode allows the throughput of DMS-TCP to reach the high speed region quickly. Therefore, DMS-TCP supports scalability. We describe an example for explaining the start-up mode as follows: This example assumes that RTT is 100 ms and the measurement time is 2 sec. After the n -th control period, total window size is as Eq. (6). When n is 5 ($Th_{sus} = 16$), window size is 1140. When n is 6 ($Th_{sus} = 32$), window size is 2400.

$$W_{total} = \frac{T}{RTT} \cdot \left\{ n \cdot (2^{n-1} - 1) - \sum_{a=1}^n a \cdot 2^a \right\} \quad (6)$$

If a packet size is 1500 bytes, DMS-TCP has throughput of 136 Mbps and 288Mbps after 5th and 6th control periods, respectively.

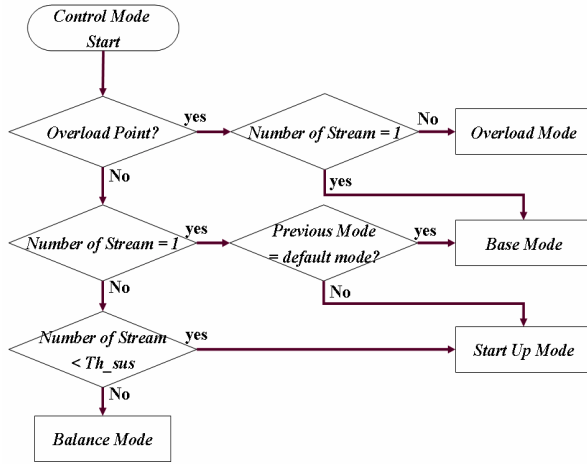


Fig. 3. Flow chart for selecting four modes

Balance Mode. The control module enters into balance mode after the number of streams reaches Th_{sus} . This mode would add one stream every control period as long as the OP does not occur.

Overload Mode. If the OP is detected after the measurement period, then Overload mode operates. The operation of this mode deals with congestion by decreasing the number of streams. Here, the decreased number of streams is the product of the current number of streams and DR.

4 The Properties of DMS-TCP

The High BDP protocols improve the performance of scalability by modifying the TCP's AIMD. As mentioned above in section 2, fairness and friendliness as well as scalability should be considered as criteria to test these protocols. In this section, DMS-TCP is analyzed in terms of scalability, fairness and friendliness, as compared with HSTCP.

4.1 Scalability

This paper defines scalability as the throughput of the protocol in adapting to high speed bandwidth. In [1], the HighSpeed response function equals $W = 0.12/p^{0.835}$. The average congestion window of standard TCP equals $1.2/\sqrt{p}$. For N parallel TCP connections, the aggregate congestion window for N connections equals $N \cdot 1.2/\sqrt{p}$. From the HighSpeed response function and the relationship above, we can derive the following:

$$N(w) = 0.23 \times w^{0.4} \quad (7)$$

For $N(w)$, the number of parallel TCP connections emulated by the HSTCP response function, and for $N(w) \geq 1$. From this relationship, 10Gbps throughput requires 23 TCP streams or a 10^5 congestion window size of HSTCP. This is shown in table 1 [1]. The most important consideration for scalability is how long it takes to obtain target throughput or the number of streams. In DMS-TCP start-up mode, the number of streams doubles every $T_m + T_c$ (cycle). Accordingly, T_m is the variable for scalability. The short T_m causes inaccurate measurement results, but fast increment of the number of streams. For example, when Th_{sus} is larger than 32, DMS-TCP satisfies the condition of 10Gbps after 6 cycles.

Table 1. Number of $N(W)$ of parallel TCP connections roughly emulated by the HighSpeed TCP response function

CWND (HSTCP)	100	1000	10000	100000
Number $N(W)$	1.4	3.6	9.2	23.0

4.2 Fairness

Fairness means that each end-to-end connection achieves an equal share of the bandwidth, if there is more than one connection sharing a bottleneck link. If some senders transmit the data at the same time, DMS-TCP is equal to the number of streams as the senders, although different RTTs. As RTT is smaller, the amount of transmitting packets and packet losses are larger. Thus, if any sender has a smaller RTT, this would suffer from greater congestion. This also means that the sender with a small RTT has a lower number of streams then the sender with a large RTT. That is, it is possible to provide fairness by differentiating the number of streams in DMS-TCP.

4.3 Friendliness

Friendliness indicates that a new protocol does not affect legacy TCP in terms of throughput in any condition. Accordingly, standard TCP should obtain equal throughput for DMS-TCP in low bandwidth conditions. If congestion occurs in low bandwidth, DMS-TCP would act in base mode, in which there is only one stream. Thus, DMS-TCP acts as a standard TCP since the base mode maintains a single stream.

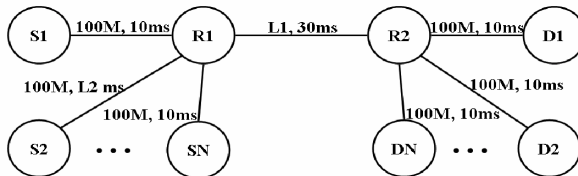


Fig. 4. Flow chart for selecting four modes

The High BDP protocols improve the performance of scalability by modifying the TCP's AIMD. As mentioned above in section 2, fairness and friendliness as well as scalability should be considered as criteria to test these protocols. In this section, DMS-TCP is analyzed in terms of scalability, fairness and friendliness, as compared with HSTCP.

5 Performance Evaluation of DMS-TCP

In this section, the performance of DMS-TCP is evaluated through NS-2 simulation, as compared to standard TCP and HSTCP in terms of scalability, fairness and friendliness. This simulation is accomplished using an ns-2.28 version on SunOS 5.8. A network model for simulation is a simple dumbbell topology (Fig. 4). The fixed parameters for this model are as follows: DR (50%), T_m (2 s), T_{h_sus} (8) and OPL (the same as the number of streams). The other parameters of the number of nodes, bottleneck bandwidth, RED queue buffer size and link delay are varied to compare the performance of DMS-TCP and HSTCP.

5.1 Scalability Evaluation

The first experiment is to test scalability. Accordingly, several simulations are performed in different bottleneck bandwidth (L1) of topology shown in fig. 4. Further, the topology opens two end-to-end connections for each simulation. Three simulations are performed using DMT-TCP, HSTCP and standard TCP, respectively. The results of this experiment are shown in Fig. 5, in which average throughput achieved by DMT-TCP, HSTCP and standard TCP are compared. From Fig. 6, it is shown that both HSTCP and DMS-TCP perform well as the bottleneck bandwidth grows. However, TCP Reno shows the most significant degradation. Furthermore, DMS-TCP shows 12% improvement over HSTCP's throughput in 10Gbps.

5.2 Fairness Evaluation

The second experiment is performed to evaluate the fairness property of DMS-TCP. In the simulation topology, the L2 link delay between S2 and R1 is considered as 10, 35, 60, 85, and 110ms and L1 link bandwidth between R1 and R2 is fixed to 500M. This paper uses Jain's fairness index [12].

Fig. 6 shows that HSTCP is almost inversely proportional to RTT^2 , and DMS-TCP is almost inversely proportional to RTT . HSTCP refers only current CWND size when it calculates increment size. It causes HSTCP to suffer from serious unfairness problems. However, if DMS-TCP senders start at the same time, all senders would have the same number of streams in spite of RTT difference. Thus, we conclude that DMS-TCP is no less fair than HSTCP.

5.3 Friendliness Evaluation

In this experiment, DMS-TCP and standard TCP connections coexist in the simulation topology, with low bottleneck link L1 bandwidth: 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100 Mbps. This experiment observes whether standard TCP operates well against DMS-TCP in low speed network conditions.

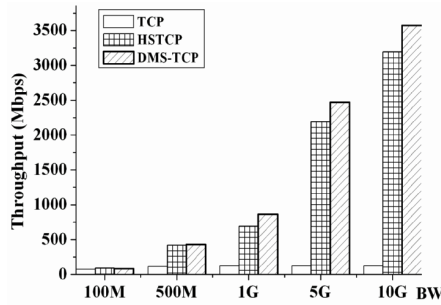


Fig. 5. Scalability Result

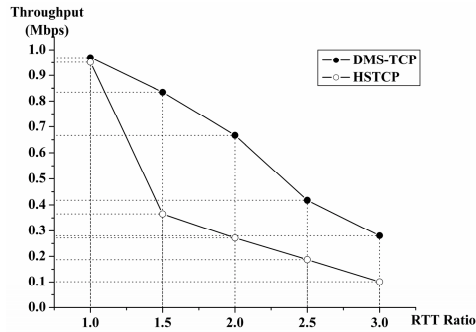


Fig. 6. Fairness Result

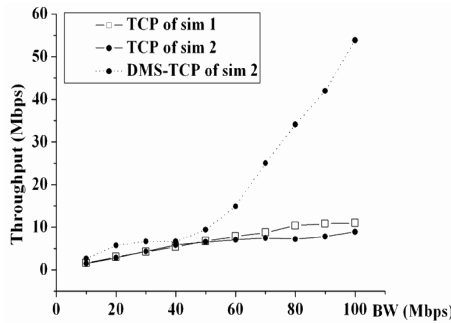


Fig. 7. Friendliness Result

There are two simulations, in simulation1, only one TCP connection is opened, but in simulation2, a TCP connection and DMS-TCP connection coexist. Therefore, it is possible to evaluate the friendliness by comparing ‘TCP of sim1’ with ‘TCP of sim2’. The throughput of ‘TCP of sim2’ is similar to that of ‘TCP of sim1’. This means that DMS-TCP does not affect the throughput of TCP. In the case of greater than 60Mbps, DMS-TCP obtains further bandwidth than TCP. This is because TCP’s AIMD

mechanism cannot support greater than approximately 15Mbps in this simulation case. From the results of this experiment, it is concluded that DMS-TCP provides better performance in terms of friendliness.

6 Conclusions

This paper proposes DMS-TCP, using multiply streams over a single end-to-end path, which controls the number of streams for scalability. DMS-TCP considers fairness and friendliness, in addition to scalability. The performance of DMS-TCP is evaluated through the NS-2 simulation, as compared to standard TCP and HSTCP, in terms of scalability, fairness and friendliness. Simulation results demonstrate that DMS-TCP can effectively deliver the desired performance in high speed networks. Further work is still required to consider a wide range of background traffic scenarios, subject to dynamic flows and network changes under more realistic conditions.

References

1. Floyd, S.: HighSpeed TCP for Large Congestion Windows. IETF RFC 3649 (2003)
2. Floyd, S., Ratnasamy, S., Shenker, S.: Modifying TCP's Congestion Control for High Speeds. Internet draft (2002), <http://www.icir.org/floyd/notes.html>
3. Dalton, L.A., Isen, C.: A study on High Speed TCP Protocols. In: IEEE GLOBECOM, vol. 2, pp. 851–855 (2004)
4. Altman, E., Barakat, C., Laborde, E.: Fairness Analysis of TCP/IP. Decision and Control 2000, vol. 1, pp. 61–66 (2000)
5. Kelly, T.: Scalable TCP: Improving Performance in HighSpeed Wide Area Networks. ACM Computer Communication Review 33, 83–91 (2003)
6. Katabi, D., Handley, M., Rohrs, C.: Internet Congestion Control for High Bandwidth-Delay Product Networks. In: Proceedings of the ACM SIGCOMM, pp. 43–54 (2002)
7. Jin, C., Wei, D.X., Low, S.H.: Fast TCP: motivation, architecture, algorithms, performance. In: IEEE INFOCOMM, vol. 4, pp. 2490–2501 (2004)
8. Hsieh, H.-y., Sivakumar, R.: Parallel Transport: A New Transport Layer Paradigm for Enabling Internet Quality of Service. IEEE Communication Magazine, 114–121 (2005)
9. Hacker, T.J., Noble, B.D., Atthey, B.D.: Adaptive Data Block Scheduling for Parallel TCP Streams. In: IEEE HPDC 2005, pp. 265–275 (2005)
10. Brendan, C., Traw, S., Smith, J.M.: Striping within the Network Subsystem. IEEE Network, 22–32 (1995)
11. Brakmo, L.S., Peterson, L.L.: TCP Vegas: End to End Congestion Avoidance on a Global Internet. IEEE JSAC 13, 1465–1480 (1995)
12. Keshav, S.: REAL: A network simulator, Department of Computer Science, UC Berkeley, Technical Report 88/472 (1988)
13. The ns-2 Network Simulator, <http://www.isi.edu/nsnam/ns/>

A Study on the Service Quality Management Process and Its Realization Strategy for Capturing Customer Value

Eunjoo Kwak, Byeong-Yun Chang, Daniel Wonkyu Hong, and Byungdeok Chung

KT Network Technology Lab.
463-1, Jeonmin-dong, Yuseong-gu,
Daejeon, South Korea
{ejkwak, bychang, wkhong, bdchung}@kt.com

Abstract. In the work presented here, we introduce the NGOSS (Next Generations Operations Systems and Software) and eTOM (Enhanced Telecom Operations Map) and study the service quality management process with eTOM. We propose guideline of service quality management business process by drawing process flow diagrams among eTOM L2 and L3 processes. Then to make a standard functional model for developing OSS supporting service quality management, we derive primary necessary functions for service quality management and resource performance management process elements. With the result of this research and our experience of operations and management on a telecommunication network, we propose improvement directions of service quality management and its implementation strategy.

Keywords: Service quality management, resource performance management, eTOM, business process, implementation strategy, OSS.

1 Introduction

Today's telecommunication operators are facing with growing fierce competitions and dramatically changing market environments due to the development of technology. The influencing power of customer is becoming stronger and their needs are getting complex and diverse [1]. In the view of service, telecommunication services are also evolving from traditional voice-centric and network-based connectivity services to convergent services based on All-IP converged network. In accordance with these telecom market and business environments, the network operations and management infrastructure is also migrating into "service-oriented" and "business-oriented" network operations and management infrastructure from "stability-oriented" network operations and management infrastructure within the context of today's multimedia and converged business environments [2].

In this competitive marketplace of modern telecommunications, gaining and retaining customers is essential to the success of telecommunications operators. Service providers are struggling to deliver Quality of Service (QoS)-guaranteed service to meet their customer's requirements and improve customer satisfaction in the battle to

win the market share. Therefore, for service providers, QoS management for meeting their customer’s needs and providing high quality service is becoming increasingly important and most operators are providing SLA. Especially, in the paradigm shift from network service provider to multimedia service provider such as Telco’s quadruple play service (QPS), it becomes more iFmportant issues ‘how to define and measure the quality of experience (QoE) that customer feels and how to measure and manage for QoS’.

For this important topic, service quality management, Tele Management Forum (TM Forum) proposes SQM process in the Assurance process in eTOM (enhanced Telecom Operations Map). In this work, we will study SQM process using eTOM and propose reference business process model and functional model and its implementation strategy

The organization of this paper is as follows. In Section 2, we briefly introduce NGOSS and eTOM. Using eTOM, we analyze process elements, draw the service quality management business process flows, and derive primary necessary functions for SQM process in section 3. Based on the result of section 3 and our experience on network operations and management, section 4 presents the improvement directions of SQM and its realization strategy and conclusions are given in Section 5.

2 NGOSS and eTOM

In this Section, we briefly introduce the TM Forum NGOSS and eTOM.

NGOSS is a comprehensive, integrated framework for developing, procuring and deploying operational and business support systems and software. NGOSS uses a common business process map, systems descriptions, and information models and couples them with pre-defined integration interfaces, architectural principles and compliance criteria [3].

Figure 1 provides a representation of NGOSS Framework, and the vital role of the eTOM framework within this. ETOM framework provides the Business Process Map for NGOSS.

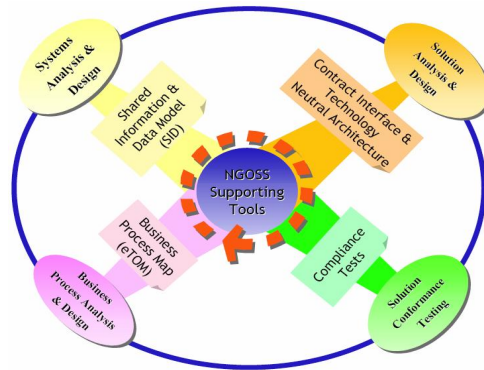


Fig. 1. TM Forum NGOSS Core Framework

The eTOM describes all the enterprise processes required by a service provider and analyzes them to different levels of detail according to their significance and priority for the business. For now, the maximum detail level of eTOM process is in level3 and it explains core business artifacts for each process area.

Figure 2 shows the Operations portion of the eTOM framework decomposed into the OSR (Operations Support & Readiness) vertical end-end process grouping plus the three Customer Operations vertical end-end process grouping of FAB(Fulfillment, Assurance, Billing). Even though the end-to-end process breakdown in Figure 2 provides a sound image of which component process belongs to which end-to-end process, it does not provide end-to-end process flow required to support Customer Operations process. Therefore, we need to analyze process elements' role and functions and design end-to-end business process, for example, Fulfillment, Assurance, Billing and our concern, service quality management.

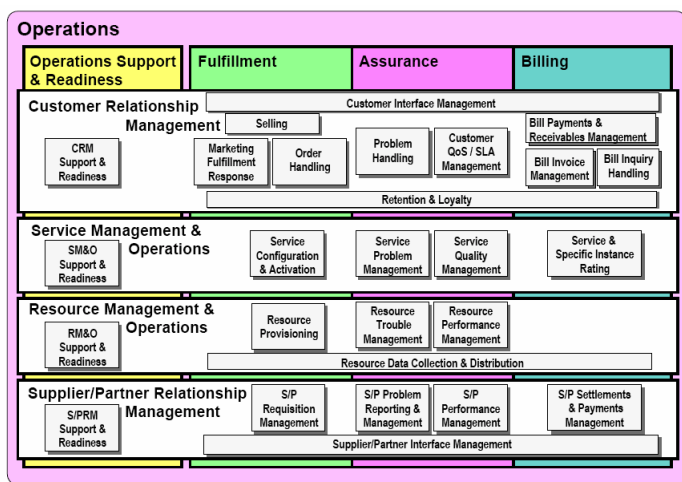


Fig. 2. eTOM Level 2 Operations (OPS) Process

In this paper, we will study and analyze focusing on eTOM Level2 Processes, Service Quality Management (SQM) and Resource Performance Management (RPM) process in the Assurance process grouping for designing the SQM business process. For more detail explanation of eTOM, you can refer the reference [3, 4].

3 Analysis of Service Quality Management Process

In this section, we design rough core business process flow by drawing eTOM L2 business process flows. Then for more detail business process that can be the guideline of business process for real business enterprises, we propose the eTOM L3 process flows for SQM and RPM process area. In Addition for developing OSS supporting service quality operations and management, we derived the standard reference functions for SQM and RPM process elements.

Before detail analysis and design of SQM and RPM processes, to understand overall flow of service quality management, we draw end-to-end process flow of eTOM L2 processes by analyzing each eTOM process' role and responsibilities. SQM process mainly consists of two ways. One is proactive process that is monitoring network resources and service resources periodically, discovering the degradation of quality, and improving the quality. The other is reactive process that is from the VOC (Voice of Customer) and searching quality information in real-time and analyzing the cause of degradation of quality and improving it confronting the VOC of quality.

For example, eTOM L2 process flow for service quality management proactive approach can be drawn as figure 3. This figure shows the overall end-end flow of service performance management proactive approach that is initiated from performance data from NMSs or passive probes then conducts the analysis, improving and resolving the trouble. This flow is just a rough guideline and it explains general core business flows among L2 processes. Therefore, this process flow of L2 process will be more detailed into Level 3 process flows in the following work as in figure 5, figure 6.

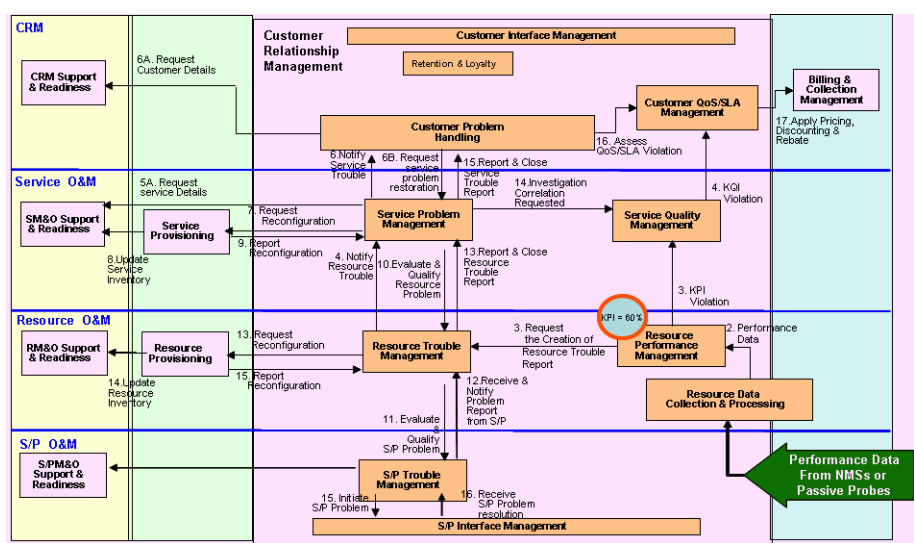


Fig. 3. Performance Management with Collected Service/Resource Monitoring - Proactive Approach (eTOM L2 Process flow)

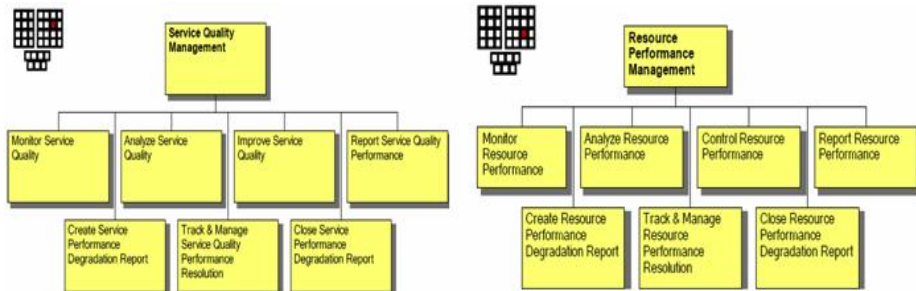


Fig. 4. SQM and RPM Decomposition into Level 3 Processes

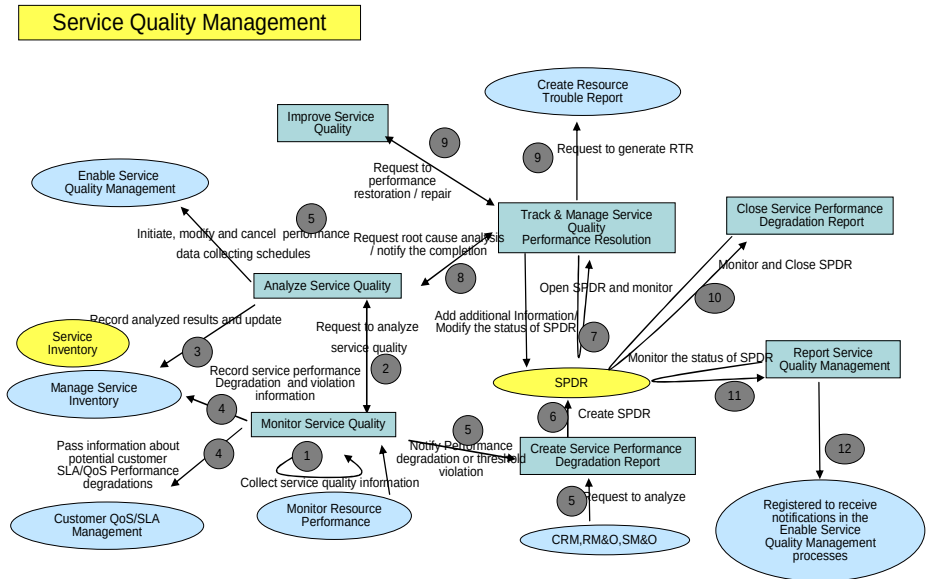


Fig. 5. eTOM Level 3 Process Flow Diagram: Service Quality Management

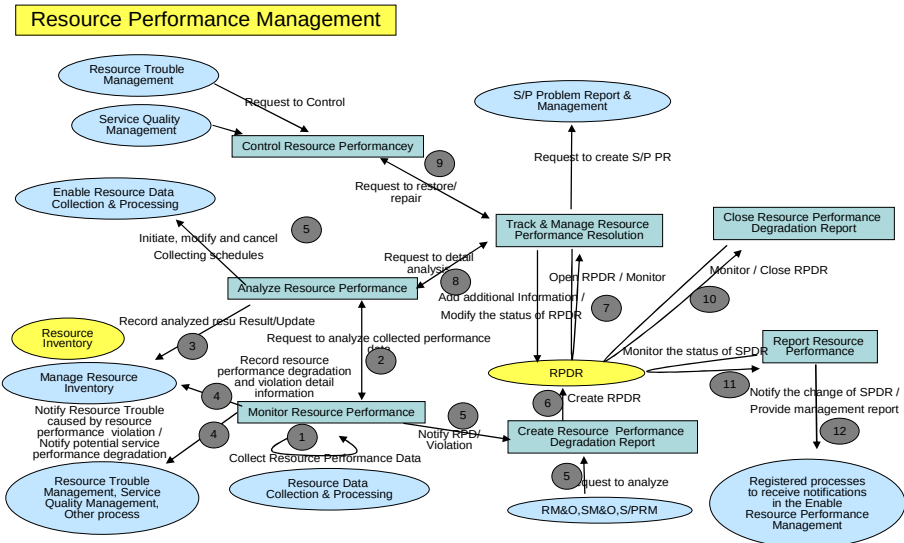


Fig. 6. eTOM Level 3 Process Flow Diagram: Resource Performance Management

Decomposition of eTOM Level 2 Processes, SQM and RPM, into Level 3 processes is shown in Fig 4 [4].

For presenting the guideline of SQM process that based on the industry standard, we first analyze each process element's object and responsibility and draw the process

flow diagram of Level 3 Process Elements, which is in the part of SM&O (Service Management & Operations) and RM&O (Resource Management & Operations) Layer of the Assurance Process. This work helps understanding of the relationship and the flow among these Level 3 process elements.

Figure 5 and Figure 6 show eTOM Level 3 business process flow of SQM and RPM, eTOM Level 2 processes. Here, square box represents that this process element belongs to its own process, and the oval shape element represents that this process element belongs to the other process. In the middle of the picture oval shapes titled as SPDR and RPDR, represent Service Performance Degradation Report and Resource Performance Degradation Report. Brief explanation of process flow of figure 5 is followed.

1. Collect and log the service quality information data
2. Request to analyze service quality information
3. Record the results of the analysis and intermediate update in the Service Inventory
4. Record service performance degradation and detail violation information / Pass information about potential customer SLA/QoS performance degradations
5. Initiate, modify and cancel continuous performance data collection schedules for analysis of service performance / Notify performance degradation or threshold violation
6. Create Service Performance Degradation Report(SPDR)
7. Open SPDR and monitor
8. Request to Root Cause Analysis of SPDR
9. Assign and coordinate service performance restoration activities and /or repair activities delegated to other process / Generate the respective resource trouble report creation requests to 'Create Resource Trouble Report' where analysis the root cause is related to resources:
10. Monitor the status of all open SPDRs and close SPDR process when the status is changed to cleared
11. Monitor the status of SPDR
12. Provide notifications of change of SPDR and management report to other processes registered in the Enable Service Quality Management processes to receive notifications of any status change

The process flow of Resource Performance Management shown in Figure 6 also can be explained as the similar flow of Figure 5.

By researching and utilizing the methodology and approach in the previous research for building an advanced OSS architecture and fault management process[5, 6, 7], we drew the figure 5 and figure 6 to understand the process flow of service quality management in SM&O and RM&O Layer. To make a complete end-to-end (from customer to suppliers and partners) process flow of service quality management, the same work on CRM and Suppliers and Partners Operation and Management (S/P O&M) layer needs to be done.

To make standard recommendation functions for building OSSs supporting service quality management, now, we derive primary necessary functions for process elements. The procedure to derive primary necessary functions is after analysis on the role and responsibility of SQM and RPM process elements in depth using eTOM and

then extracting the primary necessary functions, which are considered essential functions for developing actual OSS systems supporting service quality management operations. Table 1 shows the primary necessary functions extracted in SQM and RPM process elements. These functions can be used by standard reference functional model for improving and building OSS supporting SQM.

Table 1. Primary Necessary Functions for SQM and RPM process elements

Service Quality Management		Resource Performance Management	
Process Elements	Primary Necessary Functions	Process Elements	Primary Necessary Functions
Monitor Service Quality	Monitor service quality information	Monitor Resource Quality	Monitor resource performance information
	Filter collected data		Filter collected data
	Compare threshold and classify		Compare threshold and classify
Analyze Service Quality	Monitor service performance		Notify to other processes
	Analyze the root cause of SPD	Analyze Resource Performance	Analyze the root cause of RPD
	Filter alarm		
Improve Service Quality	Analyze service quality index	Control Resource Performance	Control to restore resource instance to normal operation and to restore failed resource
	Remote control		Notify the result of restore
	Manage improvement plan		
Track & Manage Service Quality Performance Resolution	Notify the result of restore	Track & Manage Resource Performance Resolution	Monitor RPDR
	Monitor SPDR		Search/update/add/change status in RPDR
	Search/update/add/change status in SPDR		Schedule, assign, and coordinate restoration and repair activities
Report Service Quality Performance	Schedule, assign, and coordinate restoration and repair activities	Report Resource Performance	Provide progress information of RPDR
	Provide progress information of SPDR		Statistical functions of RPDR
	Statistical functions of SPDR		Notify completion of RPDR
	Notify completion of SPDR		Notify status change of RPDR
	Notify status change of SPDR		Provide resource performance management report
Close SPDR	Provide resource fault management report	Close RPDR	Monitor RPDR
	Monitor SPDR		Close RPDR
Create SPDR	Close SPDR	Create RPDR	Create RPDR
	Create SPDR		Modify/cancel RPDR
	Modify/cancel SPDR		

4 Service Quality Management Improvement Directions and Implementation Strategy for SQM Solutions

In this section, we analyze the current SQM process and OSS status of KT and then propose the improvement direction and strategy for its realization.

First, for building improvement direction of SQM we investigate and analyze the KT's as-is service quality management process and OSSs. We analyzed each OSS' role and functions by system documents. In addition, we investigated as-is SQM process by documents and interviewing operators in field.

As this analysis, the following 2 main issues were derived.

A lack of end-to-end SQM business process definition:

In some services, probes and systems are collecting the service quality information. However, there is no following process of analysis on the collected quality information and restoration and improvement of quality degradation.

Absence of SQM business process supporting systems:

Partially there are some passive probes and systems, which measure and collect the service quality information in RPM area. However, actual support for SQM (eTOM L2 process) area is absent relative to the eTOM Operations area. It is assumed that this absence was caused by traditional 'network = service' [7] business structure. However, in the change of network-centric, service provider-centric business environment to customer-centric, business-centric one, the SQM is the key factor for the success of business.

We may think that the easiest way to solve above issues is just installing SQM solutions. However, SQM is not easy as much as most solutions say. Now we discuss some challenges for Service Quality Management.

SQM is not simply a matter of installing new equipment and SQM solutions but harmonization activities between installing new equipment and SQM solutions and a business renovation. Moreover, SQM is becoming the driving force uniting network operations, customer care, enterprise account management, and the other departments around the profitable delivery of consistent quality service to customers.

Therefore, we emphasize that before simply adopting SQM solutions, the prerequisite for service quality management is to be prepared together for successful SQM. The prerequisite I³ for service quality management is as follows.

- Business Innovation Index

Service quality management must go with business innovation – i.e. business process management and process innovation

- Customer Satisfaction Index

Customer Satisfaction Index is supporting the customer-centric business management by presenting each enterprise's competitiveness and positioning in market and the services and products needed to be improved which are causing dissatisfaction from customers.

- Service Quality Index

For establishment of SQM business process, most of all, service quality index is needed. In most service operators, research for service quality index is progressing. However, the more research and measurement for searching the optimal SQI for each enterprise needed to go on.

The combination of prerequisite I³ and the SQM solutions can be successful Telco's SQM foundation.

Now, we consider the implementation strategy of the SQM solution based on this research and KT's experience on network operations and implementation of OSSs. We can establish the implementation strategy for SQM solutions as follows.

1. SQM solution is complied with NGOSS principle and eTOM process and support its standard core system functions derived in Section 3.

In the view of eTOM, our SQM solution is to focus on the Customer QoS/SLA Management and SQM L2 process relatively RPM process because RPM can be regarded as implemented by various traditional performance/ fault management systems (i.e. NMSs) and several passive or active probes.

2. SQM solution supports as-is OSSs (Billing, Service Assurance, NMSs and etc.) and enforce the interoperation functions among them to provide seamless end-to-end SQM solution.

In other words, SQM solution uses the current performance management systems, fault management systems and probes, collects management metric in the view of network/equipment, analyzes its correlation, and derives metric in the view of customer.

3. SQM solutions can confront with future convergent service, managing various network and service not just managing specific network or single services.

Even though our target model covers now and future convergent service and network, for reducing risk of failure, we propose the incremental improvement. For example, before expanding all services we will focus on modeling 1 or 2 services and then after proofing that, expand to other services incrementally.

After analyzing the legacy OSS' role and functions, KT's target SQM solution model is show in Fig 7.

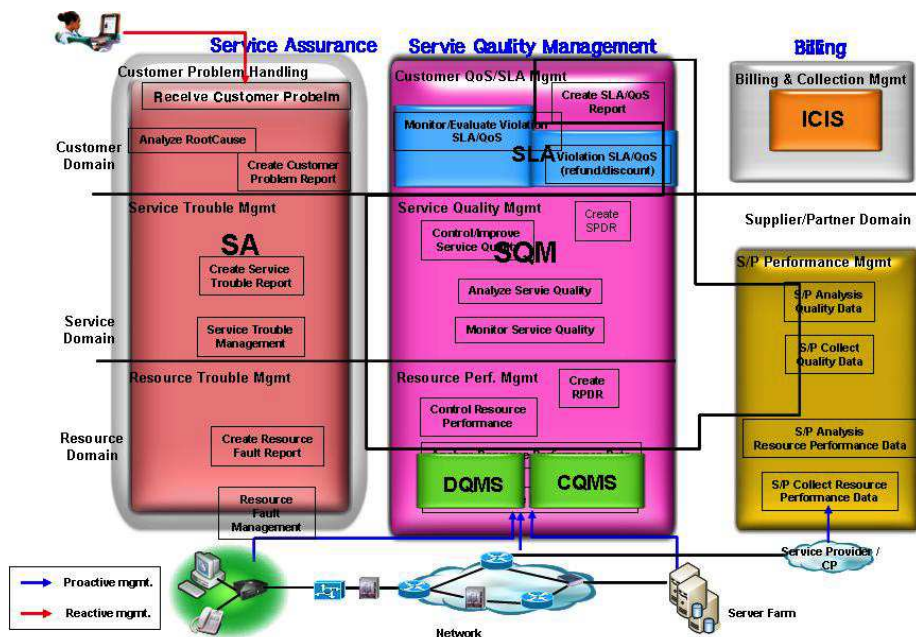


Fig. 7. KT's Target SQM Solution Model

5 Conclusion

Service Quality Management is challenging and developing requirements for today's telecommunications service operators. It becomes essential factors to meet their customer's needs and deliver QoS guaranteed services for the success in the competitive market environments.

This paper analyzed the process of service quality management based on the NGOSS eTOM framework and proposed the guideline of business process flow for service quality management by drawing process flow diagrams and derived primary necessary functions. By comparing and analyzing the proposed guideline of SQM process and derived recommending standard functions and KT's as-is SQM process and OSS status, we considered the main issues on SQM operations and management. In addition, for the implementation of SQM solutions, we discussed challenges of service quality management, and then we proposed the strategy for the implementation of SQM solutions.

Based on this study we will continue to work on defining the optimized enterprise SQM process and we will also research on definition of Service Quality Index (KPIs, KQIs, CQIs) and SQM solutions prototyping for evolving the KT's SQM capability.

References

1. Ok, K.-S.: OSS Standardization Trends: KTR&DZine, Vol.6. KT (2007)
2. Bang, Y.-H.: NOM 2.0: Innovative Network Operations and Management for Business Agility. IEEE Communications Magazine (March 2008)
3. Forum, T.M.: Enhanced Telecom Operations Map (eTOM). GB921, Release 7.0 (2007)
4. Forum, T.M.: Enhanced Telecom Operations Map (eTOM). GB921D, Release 7.0 (2007)
5. Chang, B.-Y.: A Study on Service Problem Management and Resource Trouble Management on a Telecommunication Network. In: APNOMS 2007 (2007)
6. Chang, B.-Y.: Next Generation Trouble Management Operations and Management. KNOM Review 10(1), 32–42 (2007)
7. Moon, S.-B.: Studies on Advanced OSS Architecture for Network Management in KT. In: APNOMS 2007 (2007)

A Novel Integrated Supporting System for Mesh-Pull Based P2P IPTV

Bo Wen^{1,2}, Feng Liu^{1,2}, and Luoming Meng³

¹ Network Management Research Centre, Beijing Jiaotong University

² Engineering Research Center of Network Management Technology for High Speed
Railway, Ministry of Education, 100044 Beijing, P.R. China

³ State Key Laboratory of Networking and Switching Technology, Beijing University of
Posts and Telecommunications, 100876 Beijing, P.R. China
wenbo008@gmail.com

Abstract. This paper presents a novel integrated supporting system for the mesh-pull based P2P IPTV. The proposed system can monitor the service quality of each overlays within the system by initiating random walks to sample key metrics of the system periodically and analyze the service quality statistically, and improve them to some extent. When the service quality is below the threshold, the system will increase its upload bandwidth adaptive to the workload of the system. The supporting system will allocate its bandwidth among multiple overlays to get the as much revenue as possible. This paper presents the design of this architecture and describes its functions and shows its adaptability to the high dynamic nature of p2p IPTV system and fitness for commercial deployment. At last we verify the feasibility of the architecture by simulation, and the results show its performance and adaptability.

Keywords: IPTV, Overlay, Multicast, Mesh, Monitor.

1 Introduction

IPTV which can provide live media streaming, time-shifted TV and VOD services has emerged as one of the most important services for the telecom industry[1]. Streaming video over IP networks makes great pressure to the network bandwidth and takes a lot of challenges. IP multicast can significantly reduces the cost of network, but its deployment is slow for both technology and business reasons. In recent years, application level multicast has emerged as an alternative to IP Multicast. In this architecture peers participating in a multicast group self-organize into efficient overlay for delivering data. There has been significant efforts in recent years in designing the prototype system, improving its performances, modeling, and deploying real systems[2-4].

Most of traditional peer-to-peer systems are based on the tree or mesh architecture. Mesh architecture organizes peers into an unstructured graph, and peers exchange buffer map which indicates the ownership of specified media chunks. After getting the buffer map of the neighbors, the peer can request un-received chunks in its

interested window and at the same time send the chunks to its neighbor according to the neighbors' requests[2]. Mesh architecture is fit for the high dynamics and churn property of P2P scenarios. Most of the real p2p streaming systems deployed over Internet adopt this type of architecture [5].

Some efforts have been made for the combination of the CDN and mesh architecture[6] [7]. While in most of the existing works servers function as backbones which carry a lot of deliver work, in our article, we present a novel integrated supporting system in which servers acting as supporting nodes monitor the service quality and improve it adaptive to the workload. The supporting node initiates random walks to sample key metrics of the system periodically and analyze the service quality statistically. When the service quality is below the threshold which may mean the upload bandwidth is not enough, the supporting nodes increase their upload bandwidth adaptive to the workload. But when the peers overall have enough resources to support themselves, it reduces its workload of media delivery and exists just as an ordinary peer, meanwhile focuses on the monitoring tasks. The supporting node will allocate its bandwidth among multiple overlays to get the as much revenue as possible. As the servers will not act as backbones, the requirement for its capacity is low, so the ordinary host could be, but it's stable and works like a firefighter.

The rest of the paper is organized as follows. Section 2 describes related work. Section 3 introduces proposed the novel adaptive integrated supporting system. Section 4 shows monitor function of the integrated supporting system. Section 5 describes the supporting function of the integrated supporting system when the bandwidth is not enough. Section 6 focuses on the bandwidth allocation task among multiple overlay. We show the effectiveness of the architecture by simulations in Section 7 and Conclusions are given in Section 8.

2 Related Work

There is a lot of research work about scalable distributed overlay monitoring, while most of the works were focused on monitoring tools and frameworks. They gather specific node statistics (like degrees, bandwidth), collect events and report to trustworthy nodes, process data to some extend. Xing Jin etc. studies monitoring of peer-to-peer streaming[8], but their proposed system focused on malicious actions of peer. Xiaojun et al. [9] measure PPLive[10] by active crawling and passive sniffing technique. Their crawlers connect as many peers as possible to get the exact statistics and not fit for the real-time monitoring of dynamic large scale system. Daniel et al. [11] study the influence of dynamics and heterogeneous nature of peer to the selection of representative samples and propose a modified random walking method. But they mainly focus on the file-sharing system properties such as degrees, bandwidth and number of shared files etc.

Wu et al. [6] study the importance of servers and the optimized allocation of server capability in multi-channel and multi-ISP P2P live streaming system. Our work is different from theirs for our proposed system is just supporting system and focus more on sampling, and instead of predicting and proactively allocating capacity like [6], we compensate the lack of bandwidth passively and based on monitoring results iteratively which is more cautious. Because we don't need the support nodes to be powerful as

CDN servers do. But we admit we do get a lot of inspirations from them. We propose the novel integrated supporting system in which deployed for many purposes and can dynamically change bandwidth according to the monitoring results. We use distributed and parallel random walks to produce minimum monitoring delay and cost.

3 Integrated Supporting System Overview

The basic system is still based on mesh-pull p2p protocol. The live video content which is streamed and divided into media chunks is made available from the server. The peer who is interested in the media content, requests from the tracker server a list of peers which are watching the same channel and establishes neighbor relationships with them. Each peer exchanges buffer maps periodically and requests from its partners the chunks that it is interested.

The proposed integrated supporting system is consisted of supporting nodes (SN) which have no special requirement. They are stable, powerful, and fixed as the media server's neighbor. The origin server will choose them as neighbors as the first choice deliver the media content and the protocol between them is still mesh-pull based which can improve scalability. The supporting system can function as both powerful ordinary host in media delivery and monitoring system, adaptive to the workload of the system. The SNs are responsible for their own regions and make decisions separately, but they can communicate and help each other when the workload is not well balanced.

When the bandwidth of the nodes is not enough, the supporting nodes will function as servers with powerful upload bandwidth. The new arrival nodes will be redirected to the supporting nodes with near locality or in the same ISP to find neighbors, but they can only have one supporting node as neighbor at most.

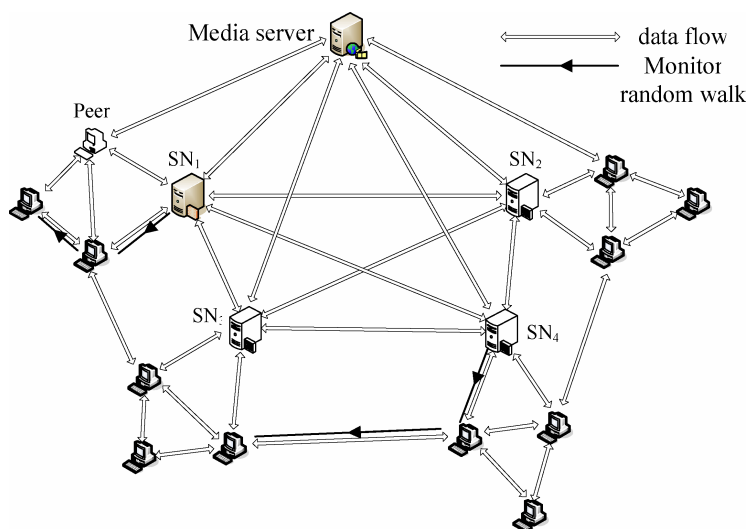


Fig. 1. The system overview

When more peers joins the system, the system will have more upload bandwidth overall and can be more tolerant to the dynamics of the peers, like departure or joining, the supporting node will decrease the bandwidth allocate to the overlay and just function as an ordinary peer.

The supporting nodes periodically collect the key metrics of the system, like lifetime, bandwidth and quality of the video of peers etc. As the exact statistics is difficult, we can monitor and estimate the results by sampling. The supporting nodes initiate random walks in the system to sample to get what it wants. The locality-aware and ISP-aware property of supporting nodes make the random walks finish in a reasonable time, and can be speeded up more by initiating parallel random walks.

When the network conditions change, e.g. the overall bandwidth decreases and the service quality may be influenced. When it is lower than the default threshold, the supporting nodes will increase their bandwidth to improve the system performances. This transition threshold is based on the monitoring results which are carried by the supporting nodes as shown (1)(2). The BW denotes the bandwidth and Quality denote the play continuity and the thresholds of change to ordinary peers are:

$$\sum_{i=1}^n BW_i^{sample} / n \geq BW_{threshold} . \quad (1)$$

$$\sum_{i=1}^n Quality_i^{sample} / n \geq Quality_{threshold} . \quad (2)$$

When there are multiple overlays in the system, which represent multiple TV programs in the system, the supporting nodes have to allocate bandwidth among them. The allocation has to assurance the priority first and then get as much as possible revenues by guarantying some level of performances. Although the performances of the overlay are decided by many factors, like sizes and overall bandwidths of the overlay, the performances may be improved if we adjust the bandwidth of the supporting nodes. And we can incorporate some access control mechanisms at the supporting nodes to guarantee the performance of the system further when new arrival nodes come and find neighbors.

4 Monitor Function

4.1 Sampling Based on Random Walk

The supporting system keeps monitoring the network-wide quality and makes the transition based on the real-time results. But to remotely monitor network-wide service quality exactly is very difficult in a large-scale dynamic and distributed mesh-pull P2P live media system.

While collecting the statistics from the log servers are delayed and lack of scalability, the method of directly measuring the quantities of interest on every peer is too expensive. Sampling is a light-weight alternate approach for collection. But it has to overcome the probable bias of common sample which is tend to the short-lived and high-degree peers due to the dynamic nature of peers[11]. We use the variation of MRWB[11] which has been verified in the unstructured file sharing system(for an

excellent survey see[12]) and just make use of the conclusion relevant to our monitoring. The exact relationship between random walk based sampling and real values of P2P live media streaming system is out of the focus of this paper, but we still can conclude that the appropriate sample is consisted with the overall system in the system level.

We adapt MRWB to suit for our situation in which all the peer has nearly the same neighbors and avoid supporting node when random walk meets it. It will just collect the bandwidth, quality of service etc and goes on to explore a neighbor of it uniformly at random. The random may visit the same peer twice or more and distinguish between occurrences at different times to be unbiased.

The transition matrix $P(x, y)$ is denoted in (3), which describes the probability of transitioning to peer y if the walk is currently at peer x , where Nbr_{set} is the default number of neighbors of ordinary peers.

$$P(x, y) = \begin{cases} \frac{1}{\text{degree}(x)} = \frac{1}{Nbr_{set}} & y \text{ is a neighbor of } x \\ 0 & \text{otherwise} \end{cases}. \quad (3)$$

And if the vector v describes the probability of currently being at each peer, then the vector $v' = vP$ describes the probability after taking one additional step. Likewise, vP^r describes the probability after taking r steps. As long as the graph is connected and not bipartite, the probability of being at any particular node, x , converges to a stationary distribution like [12] and be extended to (4):

$$\pi(x) = \lim_{r \rightarrow \infty} (vP^r)(x) = \frac{\text{degree}(x)}{2 \cdot |E|} \approx \frac{Nbr_{set}}{2 \cdot (\frac{1}{2} N \cdot Nbr_{set})} = \frac{1}{N}. \quad (4)$$

We can parallel initiates many random walk from the supporting nodes and combine them, while keeping the walk length relatively short. We choose r an appropriate value in order to reduce significant bias and meanwhile keep efficient. From graph theory[12], we expect to require $r \geq \log(|V_{peers}|)$ for sampling.

4.2 Sampling Results Analysis

We can collect and analysis the results of sampling, like bandwidth, the service quality, and may collect the buffer maps which can be used to infer more important performance metrics of the system and make decisions based on the results.

The estimation of bandwidth is based on the average bandwidth of sample nodes with a coefficient to fix the probable bias denoted in (5):

$$BW_{average}^{estimate} = b \frac{1}{N_{sample}} \sum_i^{sample} BW^i. \quad (5)$$

The service quality Q_{sum} includes the continuity index, the startup latency, the playback lags between peers and etc denoted in (6), with different coefficients for their

priorities, as ε for other influencing factors. Q_{lag} is computed with the $Q_{play-latency}$ denoted in (9), while others are just average of samples like in (7) (8).

$$Q_{sum} = c_1 Q_{continuity} + c_2 Q_{latency} + c_3 Q_{lag} + \varepsilon. \quad (6)$$

$$Q_{continuity}^{estimate} = b_1 \frac{1}{N_{sample}} \sum_i^{sample} Q_{continuity}^i. \quad (7)$$

$$Q_{play-latency}^{estimate} = b_2 \frac{1}{N_{sample}} \sum_i^{sample} Q_{plat-latency}^i. \quad (8)$$

$$Q_{lag} = \Theta(Q_{play-latency}). \quad (9)$$

The random walk method can collect buffer map information for more deep performance analysis which is not the focus of this paper, and there exists correlation between the service quality and buffer map denoted as (10).

$$Q_{sum} = g\left(\bigcup_{i=1}^{sample} BufferMap_i\right). \quad (10)$$

5 Supporting Function

Then the integrated supporting system will adapt its upload bandwidth of supporting nodes based on their own respective monitoring results. First we dynamically describe the relationship among service quality, server bandwidth, the number of peers, and the overall bandwidth of peers, denoted as the service quality function in(11)(12)(13), where Q denotes quality, BW denotes bandwidth, N denotes the system popularity, and ϕ denotes the function.

$$Q_{continuity} = \phi_1(BW_{SN}, BW_{peers}, N_{peers}). \quad (11)$$

$$Q_{latency} = \phi_2(BW_{SN}, BW_{peers}, N_{peers}). \quad (12)$$

$$Q_{lag} = \phi_3(BW_{SN}, BW_{peers}, N_{peers}). \quad (13)$$

The relationship between the goal and its influence factors is very complex. The number of peers can be obtained from the log server exactly. The bandwidth of peers overall the system can be conducted from the sampling results but not controlled. Only the supporting server node can dynamically increase its output bandwidth if the service quality degrades below the threshold, and the step to be increased is based on the absolute value of the minus between the goal and current quality denoted as (14), where function $E()$ denotes the difference, and $\psi()$ denotes the system overall peers capability. The increase will continue again and again until it achieves the required service level. The supporting system will make full use of history information and derive the estimated bandwidth change using interpolation to speed up the reactive procedure. When service quality is high enough or SN be released for other purposes, it will follow the similar way to decrease the bandwidth.

$$BW_{server} = \frac{cE(|Q_{goal} - Q_{sum}|)}{\Psi(N_{peer}, BW_{peers})}. \quad (14)$$

6 Resource Allocation Optimization

When there are multiple overlays, which represent multiple channels within P2P live media streaming system, they share the same supporting system. The supporting nodes have to allocate supporting upload bandwidth among them. The allocation is performed based on the monitoring results in a distributed manner within each overlay independently, with sum bounded by supporting node capacity.

Assuming there are M concurrent overlays in total to be supported, represented as a set O . There are N^o peers in overlay o , and $o \in O$. Let BW_{SN}^o denotes the supporting node upload bandwidth to be assigned to overlay o , and Q^o denote the service quality of overlay o , which were statistically collected from random walk sampling. Let BW_{all} be the upload bound of supporting node. We assume a price p^o for each overlay o , a priority level K^o for each overlay o , which assigned by the P2P streaming service provider to reflect the difference of prices and relative importance of the overlays.

The supporting nodes passively keep monitoring the system of each overlay and dynamically adjust upload bandwidth to compensate correspondingly to achieve optimal allocation of bandwidth, based on their prices, priorities and numbers of peers, quality of streaming service in each overlay o . The optimized problem Resource-Allocation is formally described as follows:

$$Revenue = \max \sum_{o \in O} K^o \cdot P^o \cdot N^o \cdot Q^o. \quad (15)$$

Subject to:

$$\sum_{o \in O} BW_{SN}^o \leq BW_{all}. \quad (16)$$

$$BW_{SN}^o \geq BW_{stream}^o. \quad (17)$$

$$Q_{sum} = c_1 Q_{continuity} + c_2 Q_{latency} + c_3 Q_{lag} = \Phi(N^o, BW_{peer}^o, BW_{SN}^o). \quad (18)$$

In one word, the revenue can be transformed to a function of

$$Revenue = \max \sum_{o \in O} K^o \cdot P^o \cdot N^o \cdot \Phi(N^o, BW_{peer}^o, BW_{SN}^o). \quad (19)$$

The BW_{SN}^o will not be easily allocated to improve the performance of quality of all the overlays when the relationship between the service quality function and other factors is very complex. In our supporting system, we simply heuristically adjust it by averaging which means reduce the bandwidth devoted to the highest quality overlays and move it to the most lowest quality overlays for a while considering the optimized

goal. By doing so, we can guarantee more important peers in the network can achieve satisfying streaming qualities. The upload bandwidth assignment is periodically carried out to adapt to the changing demand in each of the overlays over time.

7 Simulation

Our simulation constructs the mesh-pull P2P media streaming system using the basic mechanism of DONet[2]. We simulate 1000 nodes online, which join the system dynamically. The default media streaming rate is 300kbps and set neighbor count to 15. The data end-to-end delay is from a real-world node-to-node latency matrix used on Internet[13]. We use the term capacity index (CI), which is defined as the ratio of the total upload capacity to the video streaming rate times the receiver number. Our simulation lasts for 300sec and we began when the nodes finish buffering at about 40 seconds.

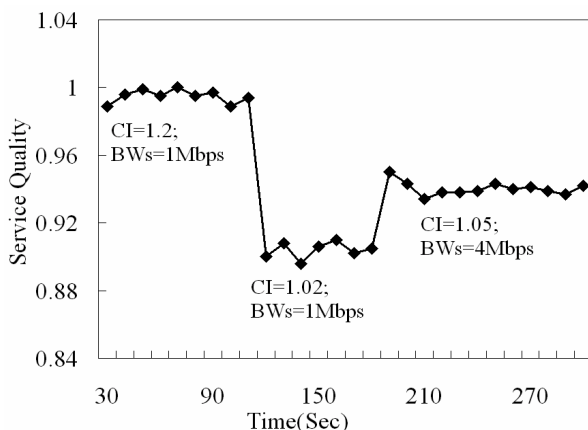


Fig. 2. The service quality using the integrated supporting system

The capacity index is set $CI=1.2$ and the supporting node is set to 1 Mbps. As the upload bandwidth of peers is enough themselves, the service quality keeps at a high level about 99%. But at time $t=100s$, the network condition changes and the bandwidth capacity index drops 1.05, the upload bandwidth is at the margin and the service quality degrades to near 90%. We keep the supporting system to hold on for a time interval to show its effect. At time $t=200s$, the supporting system functions and adapts its upload bandwidth to 4 Mbps, and the service quality was improved to 94% as shown in Fig 2. The peers join and leave the system dynamically mostly at setting time to simulate the bandwidth change, and we ignore the average quality degradation during transition until the system becomes stable. Although the bandwidth of supporting nodes contribute is increased only by 3 Mbps, which is less than the decrease of peers $1000 \times 300 \times (1.2 - 1.05) \text{ kbps} = 45 \text{ Mbps}$, the service quality is kept at a reasonable level. In real scenarios where the supporting system keeps monitoring service quality,

the supporting nodes will react quickly even before the service degrades greatly, especially considering the buffer of peers.

We compare the influence of different supporting node bandwidths on the service quality and the results is shown in Fig.3. There are 500 nodes in the system and the CI is set 1.2, and we begin the comparison after $t=100s$ when the system is stable. We can see clearly that the more the supporting node contributes its upload bandwidth, the better quality the service can achieve.

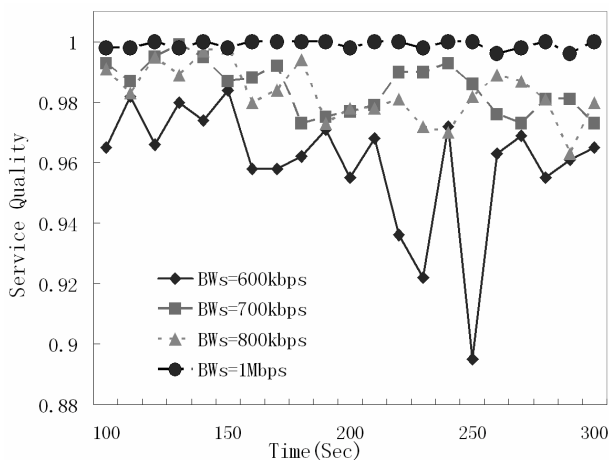


Fig. 3. The service quality with different SNs upload-bandwidth setting

8 Conclusion

This paper focus on a novel integrated supporting system in which supporting nodes not only monitor the service quality, but also improve it dynamically when needed. The supporting node initiates parallel random walks to sample key metrics of the system periodically and analyze the statistical service quality. When the service quality of system is below the threshold value, which may mean the upload bandwidth is scarce; the supporting nodes will function as powerful nodes and increase upload bandwidth adaptive to the workload of the system. But when the peers overall have enough resources to support themselves, it exhibits as an ordinary peer and focuses on the monitoring tasks. We also discuss the bandwidth allocation among multiple overlays. And the simulations show its feasibility and adaptability.

Acknowledgments

This paper is supported by the National High Technology Research and Development Program of China 2006AA01Z121.

References

1. Takahashi, A., Hands, D., Barriac, V.: Standardization activities in the ITU for a QoE assessment of IPTV. *Communications Magazine* 46, 78–84 (2008)
2. Xinyan, Z., Jiangchuan, L., Bo, L.: Yum: CoolStreaming/DONet: a data-driven overlay network for peer-to-peer live media streaming. In: *Proceedings IEEE INFOCOM 2005*, vol. 3, pp. 2102–2111. IEEE Press, Los Alamitos (2005)
3. Yang-hua, C., Rao, S.G., Seshan, S., Hui Zhang, A.H.Z.: A case for end system multicast. *IEEE Journal on Selected Areas in Communications* 20, 1456–1471 (2002)
4. Zhang, J., Liu, L., Ramaswamy, L., Pu, C.: PeerCast: Churn-Resilient End System Multicast on Heterogeneous Overlay Networks. *Journal of Network and Computer Applications (JNCA)* (2007)
5. Hei, X., Liu, Y., Ross, K.W.: IPTV over P2P streaming networks: the mesh-pull approach. *Communications Magazine* 46, 86–92 (2008)
6. Wu, C., Li, B., Zhao, S.: Multi-Channel Live P2P Streaming: Refocusing on Servers. In: Li, B. (ed.) *INFOCOM 2008*, pp. 1355–1363. IEEE Press, Los Alamitos (2008)
7. Tran, M., Tavanapong, W.: On Using a CDN's Infrastructure to Improve File Transfer Among Peers. In: Dalmau Royo, J., Hasegawa, G. (eds.) *MMNS 2005. LNCS*, vol. 3754, pp. 289–301. Springer, Heidelberg (2005)
8. Jin, X., Xia, Q., Chan, S.H.G.: Building A Monitoring Overlay for Peer-to-Peer Streaming. In: Xia, Q. (ed.) *Global Telecommunications Conference GLOBECOM 2006*, pp. 1–5. IEEE Press, Los Alamitos (2006)
9. Xiaojun, H., Yong, L., Ross, K.W.: Inferring Network-Wide Quality in P2P Live Streaming Systems. *IEEE Journal on Selected Areas in Communications* 25, 1640–1654 (2007)
10. Xiaojun, H., Chao, L., Jian, L., Yong Liu, A.Y.L., Ross, K.W.A.R.K.W.: A Measurement Study of a Large-Scale P2P IPTV System. *IEEE Transactions on Multimedia* 9, 1672–1687 (2007)
11. Stutzbach, D., Rejaie, R., Duffield, N., Sen, S., Willinger, W.: On unbiased sampling for unstructured peer-to-peer networks. In: *SIGCOMM 2006*, pp. 27–40. ACM Press, New York (2006)
12. Lovasz, L.: Random walks on graphs: A survey. *Combinatorics: Paul Erdős is Eighty 2*, 1–46 (1993)
13. Meridian node to node latency matrix (2500×2500), meridian project (2005), <http://www.cs.cornell.edu/People/egs/meridian/data.php>

BPEL Driven Policy Management of Virtualized Network Resources for IMS Environments

Nobutaka Matsumoto, Takahiro Miyamoto,
Michiaki Hayashi, and Hideaki Tanaka

KDDI R&D Laboratories, 2-1-15, Ohara, Fujimino-shi, Saitama, Japan 356-8502
{nb-matsumoto, tk-miyamoto, mc-hayashi, hide}@kddilabs.jp

Abstract. BPEL Driven Policy Manager (BDPM) designed to coordinate network resources with Web service interfaces for IMS environments is developed. BPEL workflow is applied to the policy manager for achieving end-to-end assurances of QoS and bandwidth on heterogeneous network resources. BDPM analyzes QoS-related information from IMS, and identifies appropriate policy workflow to invoke end-to-end network services by allocating resources adaptively. The resource allocation results are enforced to the distributed network-resource-management systems using service oriented architecture (SOA) through Web service interface. Using prototype implementation of BDPM, various descriptions of QoS requirements from IMS core can be adaptively processed by BDPM and end-to-end network resource management is successfully conducted.

Keywords: Policy management, BPEL, network resource management, SOA, QoS, IMS.

1 Introduction

Recently, increase of communication traffic is one of serious issues for telecommunication carriers. On the other hand, due to the emergence of flat-rate and cheap data services, revenue growth is slowing. Since differentiation between services is not sufficient in current networks, increase of network capacity is inevitable according to the increases of traffic. Therefore, to maintain the communication quality of prior services such as emergency call and premium services, while suppressing increase of network capacity, differentiation between services is very important. Differentiation between services, which is provided statically are matured [1][2][3]. However, to utilize communication equipments more efficient, communication quality for end-to-end services should be dynamically assigned on request by request.

IMS is one of the technologies for managing end-to-end communication sessions. Being coupled with resource and admission control function (RACF) [4] defining high level functional requirements of the resource management for NGN [5], it is enabled to dynamically control quality of each services on the edge of the networks. However, the scope of RACF leaves implementation issues including an additional effort to achieve end-to-end resource management and communication quality

control. Though distributed allocation of RACF to IP/MPLS network has been defined [6] to achieve end-to-end resource management, it does not support transport resource management. To cover various types of networks from layer 2 to 3, Network Resource Broker (NRB) and Network Resource Manager (NRM) architecture [7] is proposed. NRB/NRM architecture enables end-to-end service quality control over heterogeneous networks by handling virtualized and abstracted network resources through Web-services. Since the interfaces to IMS are not defined in current NRB/NRM architecture, coordination mechanism between NRB/NRM and IMS is indispensable. To introduce end-to-end network resource management by NRB/NRM into IMS environments, effective resource allocation policies for allocating adequate resources according to the service requests from IMS, are necessary.

In Web-services-based systems such as service oriented architecture (SOA), Business Process Execution Language (BPEL) [8] is used as one of the programming languages to coordinate and mediate Web-services [9]. Therefore, for Web-services-based network resource management systems such as NRB/NRM, BPEL is considered as a promising technology to be applied to policy management functions.

In this paper, BDPM (BPEL Driven Policy Manager) is introduced. BDPM decides appropriate end-to-end network resource allocation according to service requests from IMS and pre-configured resource allocation policy. The mediated resource allocation request is delivered to NRB/NRM for controlling transport equipments.

This paper is organized as follows: Section 2 introduces an overview of network resource management in IMS environments. Section 3 describes BDPM architecture including its functional components and interfaces. Section 4 introduces demonstration of end-to-end resource allocation for IMS service requests according to bandwidth allocation policy and QoS decision policy using BDPM prototype implementation. Finally, Section 5 concludes this paper.

2 Network Resource Management in IMS Environment

Fig. 1 shows the proposed resource management architecture. End-to-end network resources can be effectively managed using SOA-based orchestration techniques, and the orchestrated network resources can be exposed to application functions. Each network is virtualized and abstracted to be served as Web-services with unified management and control interfaces in the SOA-based management system, therefore, end-to-end network resource management over various communication characteristics, protocols, and management interfaces is enabled. This approach effectively provides the end-to-end network service also to the non-IMS applications, such as conventional Video on Demand and IPTV services. On the other hand, adapting end-to-end network service to the IMS core provides bandwidth-managed services to various applications. However, the common IMS interface needs policy treatment in BDPM in order to ensure appropriate quality and to improve network resource utilization. Since only allocating adequate resources which meet requests from applications does not cope with requests that exceed the capacity of equipments, it is indispensable for BDPM to have functions of decision and control which and how many resources to allocate to each request. That is to say, not only orchestration of network resources but also mediation of service requirements is necessary.

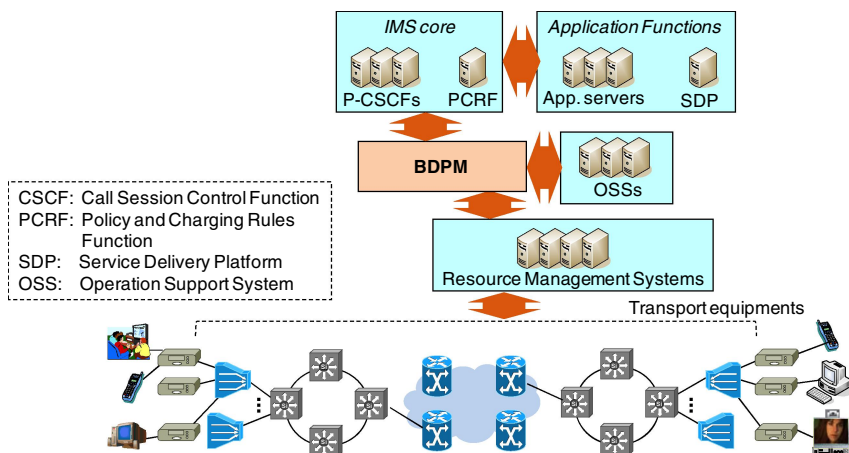


Fig. 1. Conceptual diagram of policy and resource management in IMS environments

In the proposed resource management architecture, BDPM coordinates network service requestors (e.g. IMS core, and application functions), network resource management systems and supplemental operation support systems (OSSs). By receiving requests from service requestors, BDPM decides how to handle service requests and invokes network services accordingly. The network resource management system controls network resource allocation of transport equipments to meet the decision of BDPM. As described above, infrastructure service providers such as telecommunication carriers are allowed to provide policy managed resource services to service requestors.

3 Functional Architecture of BDPM

Fig. 2 shows the functional architecture of BDPM. A primary function of BDPM is providing SOA-based end-to-end network service to the IMS core with decisions of admission and resource allocation policies. In addition to Diameter interface for the IMS core, the SOAP interface provides the network service for non-IMS application functions (e.g. application server, service delivery platform). The policy management is processed based on the work flow-based policy rules, transaction status and resource conditions. In accordance with the policy decision work flow, BDPM enforces the policy on the network resource managers through a network resource management interface. Functional block of BDPM are described below:

- **Policy engine:** Receiving request messages via Diameter [10] or SOAP service interfaces, the policy engine decides appropriate QoS parameters and operations due to a function of mediation workflow, and sends resource request messages to network resource management systems through the SOAP interface. Mediation workflow is defined using BPEL, and coordinates network resources to be allocated and operations to be driven according to the contents of request messages and resource status at that time. Examples of the operations include service initiation, service modification, service termination, etc.

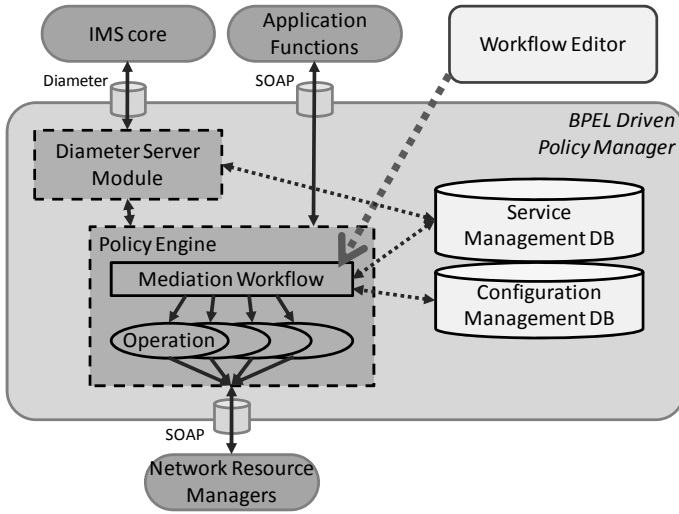


Fig. 2. Functional architecture of BDPM

- **Workflow editor:** The GUI-based policy rule editor to configure the rules defined in mediation workflow.
- **Diameter server module:** It is an extensive function block to communicate with Diameter-based service requestors such as proxy call session control function (P-CSCF) in the IMS core. Diameter is used as a base protocol to exchange resource control information with IMS core. To handle IMS Diameter interface in policy engine, Diameter engine has the role of transforming Diameter interface into a Web-service. The interface is designed for receiving service requests generated by IMS applications or servers. The specification is based on 3GPP2 Tx interface [11] using Diameter as a base protocol.
- **Service management database:** The database manages session and service information data, each including a pair of end-points, bandwidth, acceptable latency, QoS value, scheduled time, and so forth. The session information is provided to policy engine and Diameter server module as resource allocation status which is utilized in policy decision.
- **Configuration management database:** It contains policy configuration, such as conversion table used in workflow to map parameters between service requests and resource requests. Policy engine refers the database and the obtained information is utilized in resource allocation decision as a part of policies.

4 Implementation and Demonstration

4.1 Configuration

The demonstration system involves an IMS-based P-CSCF simulator, BDPM prototype, a network resource management system, and transport infrastructure as shown in Fig. 3. P-CSCF simulator communicates with BDPM by Diameter. It generates and

sends service request messages, and receives response messages from BDPM. The prototype of BDPM software is deployed on the Redhat Enterprise Linux AS4 operation system installed on Sun Fire X2100 M2 server (Dual-core AMD Opteron 1218, 2.6GHz, 4GB memory). Network resource management system consists of network resource broker (NRB) and network resource managers (NRMs). NRB provides contact point to BDPM with the Web Services Resource Framework (WSRF) [12]-based SOAP interface [13][14], and exposes end-to-end network service to each NRM providing network service within its domain. NRB is also responsible for orchestrating distributed NRMs and coordinate end-to-end network services. Each NRM is responsible for abstracting each network resource by virtualizing underlying network technologies, and manages/controls its network resources. Network infrastructure consists of GE-PON-based access networks, resilient packet ring (RPR)-based metro ring networks, and a GMPLS-based core network. BDPM intermediates the virtualized end-to-end network service to IMS core.

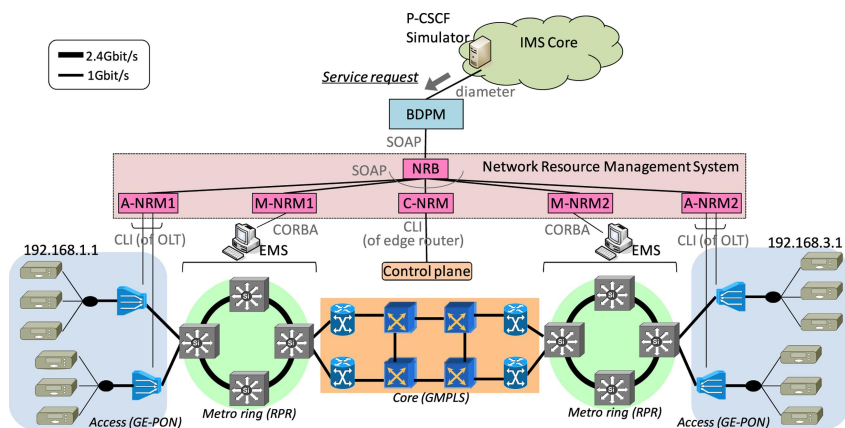


Fig. 3. Demonstration setup

4.2 Operational Sequence

Fig. 4 shows an observed operational sequence for each request. A service initiation and service termination operations are shown in Fig. 4 (a) and (b), respectively. Introducing WSRF, two-phase commit [15]-based stateful transaction was introduced to improve the synchronized operation of coordinating distributed NRMs. To achieve coherency between Diameter interface and Web services interface, AAR and RAR commands are utilized as the 1st and the 2nd phases of the reservation transaction, respectively. In service termination sequence (shown in Fig. 4 (b)), Diameter requires only one RAR/RAA sequence, while WSRF-based resource control is conducted in two-phase commitment manner. In the case of a service termination process, BDPM selects a workflow so that the policy engine runs two-phase commitment without the 2nd phase trigger from the service client.

Policy engine identifies applicable workflow and decides service parameters according to a content of the service request. Details of policy control are described in Section 4.3.

As described above, Diameter-based service requests for IMS core are fitted in Web services interface of network resource management systems, policy controlled resource management is conducted.

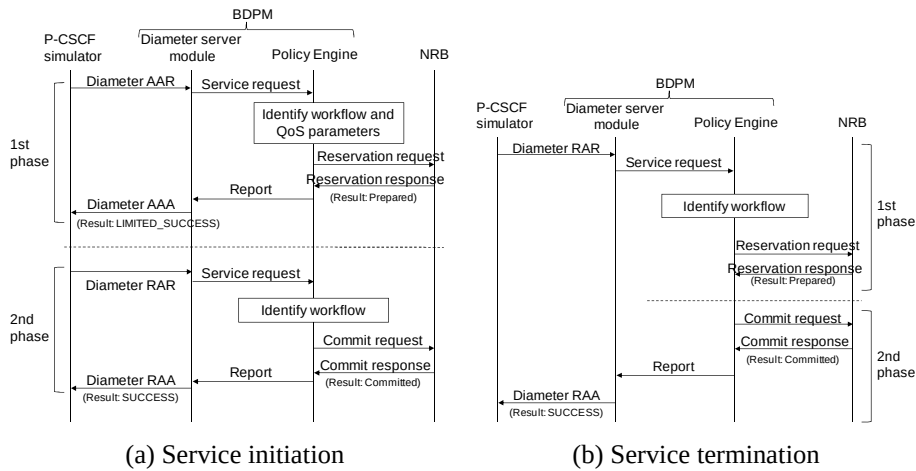


Fig. 4. Operational sequences obtained for the service initiation and the service termination

4.3 Policy Control Operation

Fig. 5 shows an overview of policy control procedure in the case of the service initiation. In the first stage (i.e. operation classification), receiving request messages from P-CSCF simulator, BDPM classifies operation of Diameter command to apply individual workflow. In the second stage (i.e. QoS policy decision), QoS policy is decided. In the second stage, high-level resource allocation policy is defined to identify QoS profiles for data flows. Since the single service initiation request also includes control flow information, control and data flows must be distinguished. In the third stage (i.e. policy enforcement), BDPM enforces a QoS and bandwidth policies of all the flows. The enforcement involves a single SOAP request message sent to NRB.

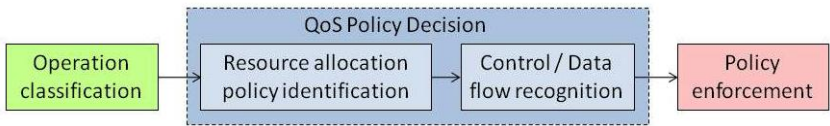


Fig. 5. Overview of operational procedure in the case of service initiation

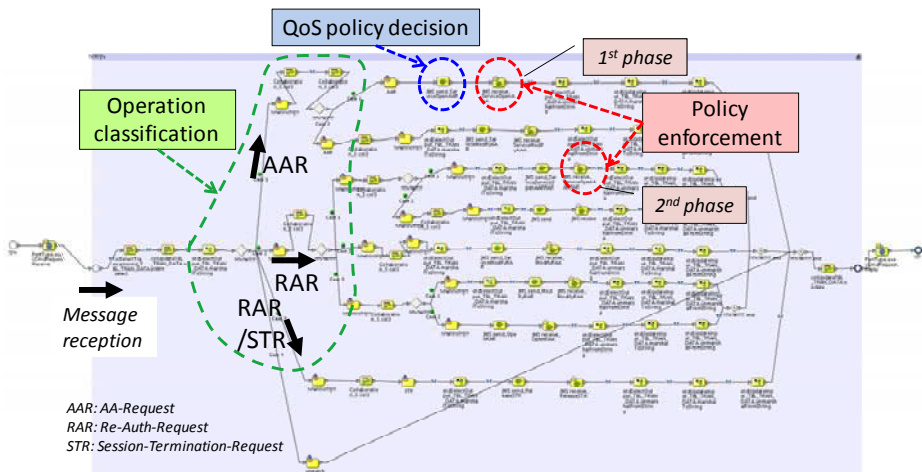


Fig. 6. Editor GUI of BPEL-based policy management workflow

Implementation of policy control operations in BPEL-based workflow is shown in Fig. 6. In the operation classification stage, the workflow path is switched considering Diameter commands (e.g. AAR, RAR, STR) and intention of the commands (e.g. AAR for service initiation, AAR for service modification). The QoS policy decision stage in the case of service initiation using AAR is shown in Fig. 7. In the case of two-phased service initiation shown in Fig. 4 (a), AAR and RAR workflow branch involves the policy enforcement stage as shown in Fig. 7.

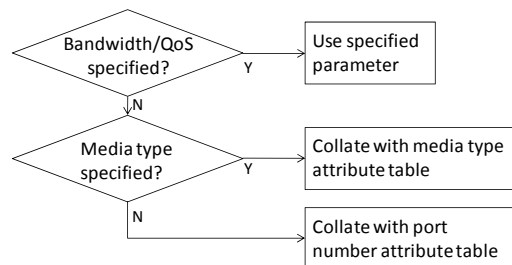


Fig. 7. A flow chart used in the resource allocation policy identification

In the first step of the QoS policy decision stage, QoS parameters for data flows are defined following a flowchart shown in Fig. 7. Explicit QoS parameters (i.e. bandwidth, QoS class) in Diameter messages are handled as primary candidate of the final QoS parameter. If the explicit QoS parameters are not specified by Diameter message, the media type information (e.g. video, voice) is referred, and the information is collated with the media type attribute table defining QoS parameters for each media types. If both explicit QoS parameter and media type information are not specified in Diameter messages, QoS parameters are defined by referring port number with collating the port number attribute table defining QoS parameters for each port types.

In the second step of the QoS policy decision stage, control flows (e.g. RTCP flow) are identified from all the media flows described in a Diameter message by using a flowchart shown in Fig. 8. By identifying the control flows, QoS parameters defined in Fig. 7 are applied to remaining data flows. The flowchart has another role of defining the bandwidth allocated for the identified control flows. In the first step of the flowchart, the existence of RTCP flow indication is verified from all the media flow information in a Diameter message. If it exists, the corresponding flows are recognized as control flows and 100 kbps is allocated to the flows. If the RTCP flow indication does not exist, then the port number of the media flows are collated with the port number attribute table providing control flow indication for each port. If the flow is identified as a control flow by the table, allocation bandwidth for the control flow is also defined by the table. If the control flows are not identified from RTCP flow indication and port number attribute table, all flows are recognized as data flows and QoS parameters defined by Fig. 7 are applied to all flows.

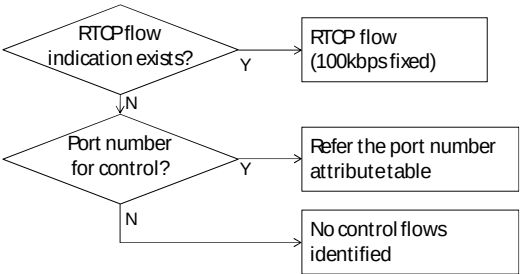


Fig. 8. A flowchart used in control/data flow recognition

Request messages used in the demonstration is described in Table 1. The request #1 contains the explicit bandwidth requirements using maximum required bandwidth of IP (i.e. data) flow. The request #2 contains implicit bandwidth requirements using media type (e.g. Audio, Video, Text). The request #3 only contains the port number related to the data flow and/or the control flow. All the three requests are composed of 4 UDP flows between 192.168.1.1 and 192.168.3.1. Detail of the 4 flows is one control channel and one data channel, both including 2 flows for each directions.

Fig. 9 shows the result of resource allocation for the three requests. Control flows of all three requests are identified using port number, since RTCP flow indication is not included. Referring the port number attribute table (Fig. 9 (b)), QoS parameters of each control flows are defined as {QoS-class=5, bandwidth=200kbps, latency=1000msec,

Table 1. Parameters of diameter request messages used in the demonstration

	bandwidth	MediaType	Port number
Request #1	100Mbps	1(video)	Ctrl:30000, Data:20000
Request #2	-	1(video)	Ctrl:30000, Data:20000
Request #3	-	-	Ctrl:30000, Data:20000

availability=0}. Latency means the maximum latency which is and availability means the function of recovery from network service trouble. As for data flows, allocated resources can be differentiated according to the QoS parameters specified in request messages. Since bandwidth requirement is explicitly specified in request #1, BDPM decides to allocate 100 Mbps for its data flow directly. Other QoS parameters (i.e. QoS class, latency, availability) for request #1 is defined referring media type attribute table. Request #2 has no bandwidth specification but has the media type information, therefore all the QoS parameters are decided by the media type attribute table (Fig. 9 (a)). Request #3 has neither bandwidth specification nor media type information, all the QoS parameters are decided by port number attribute table (Fig. 9 (b)). After the QoS policy decision, BDPM sends policy-controlled resource request messages to NRB.

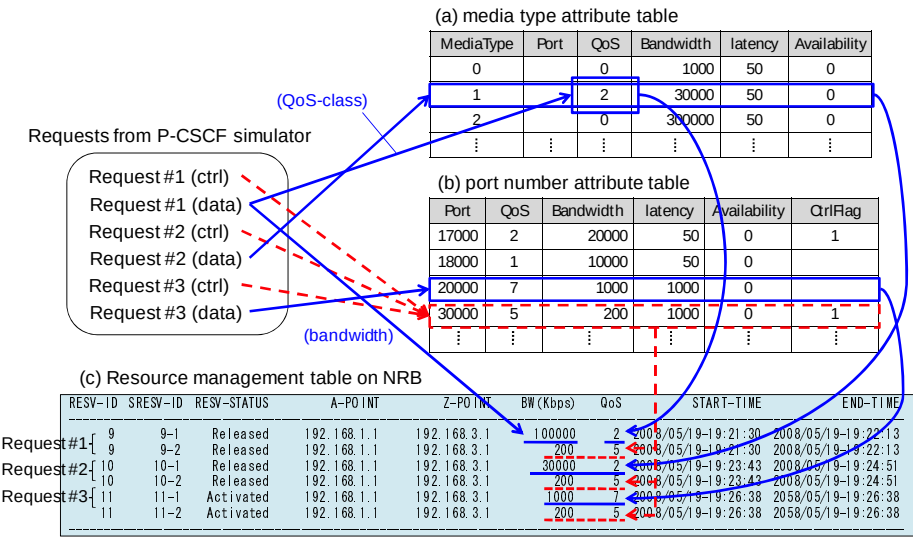


Fig. 9. Policy application and resource reservation results

NRB reserves network resources to meet the request from BDPM. The result of network resource reservation involving NRB and NRMs is shown in Fig. 9 (c). The result confirmed that BDPM successfully managed policy-based resource allocation according to the variety of requirements ranging from service-oriented requests to infrastructure-oriented requests. Since diameter-based request is not capable of specifying start time and end time, NRB reserves the requested resource instantly and pseudo-permanently (e.g. END-TIME of request #3). Start time and end time can be fully utilized by SOAP interface mainly prepared for non-IMS applications functions.

5 Conclusions

We proposed BPEL-based policy management architecture to coordinate between network resources and IMS. BPEL Driven Policy Manager (BDPM) gave the appropriate QoS conditions (e.g. QoS class, bandwidth, latency) based on implicit and

explicit QoS-related requests from IMS core considering of end-to-end assurances of QoS and bandwidth. After enforcement of the QoS conditions into distributed network-resource-management systems with Web-services interfaces, end-to-end resource management was carried out. Using prototype implementation of BDPM, BDPM processed various descriptions of QoS requirements from IMS core and end-to-end network resource management was successfully performed. From these results, BPEL workflow-based policy management is expected as an approach to provide end-to-end managed network resources for IMS environments.

In future works, scalability and performance issues will be investigated. As decentralized and distributed approaches are powerful solutions, how to coordinate distributed BDPMs, which individually handle stateful sessions, should be studied.

Acknowledgement

The authors wish to thank Dr. Shigeyuki Akiba and Dr. Masatoshi Suzuki for their continuous support for this study.

Reference

1. An architecture for differentiated services. RFC2475, IETF (1998)
2. Rajan, R., Verma, D., Kamat, S., Felstaine, E., Hersog, S.: A Policy Framework for Integrated and Differentiated Services in the Internet. *IEEE Network* 13(5), 36–41 (1999)
3. Aimoto, T., Miyake, S.: Overview of DiffServ Technology: Its Mechanism and Implementation. *IEICE Transactions on Information and Systems* E83-D(5), 957–964 (2000)
4. ITU-T recommendation: Resource and admission control functions in Next Generation Networks. Y.2111, ITU-T (2006)
5. ITU-T recommendation: General overview of NGN. Y.2001, ITU-T (2004)
6. ITU-T recommendation: Distributed RACF Architecture for MPLS Networks. Y.2174 (Y.RACF-DistribMPLS), ITU-T (2008)
7. Hayashi, M., Matsumoto, N., Miyamoto, T., Tanaka, H.: Orchestration of heterogeneous virtualized resources for end-to-end service control. In: *IEEE/IFIP NOMS* (2008)
8. Web Service Business Process Execution Language v2.0, OASIS (2007)
9. Tan, K.L.L., Turner, K.J.: Orchestrating Grid Services using BPEL and Globus Toolkit 4. In: *Proc. of 7th PGNet Symposium*, pp. 31–36 (2006)
10. Diameter base protocol. RFC3588, IETF (2003)
11. 3GPP2 recommendation: All-IP Core Network Multimedia Domain. X.S0013-013-0, 3GPP2 (2007)
12. Web Service Resource Framework v1.2, OASIS (2006)
13. Hayashi, M., Kudoh, T., Nagatsu, N.: GNS-WSI ver.1. GHPN-RG, GGF16, OGF (2006)
14. Kudoh, T., et al.: G-lambda: An interface for bandwidth reservation from applications and middleware. In: *IEEE/IFIP AGNM workshop* (2006)
15. Gray, J.: Notes on Database Operating Systems. In: Bayer, R., et al. (eds.) *Operating Systems: An Advanced Course*, vol. 60, pp. 393–481. Springer, Heidelberg (1978)

Research on the Associated Pricing Strategy in Telecom Industry

Xin Yue¹, Junjie Xu², and Zhanhong Xin¹

¹ School of Economics and Management, Beijing University of Posts and Telecommunications, Beijing 100876, China

² School of Economics and Management, Anqing Teachers College, Anhui 246133, China
angielucky@126.com

Abstract. Pricing strategy is one of the important issues in telecom industry for each player in the tri-game model. Being a conjunctive focus between users and service provider, carrier's pricing strategy takes an outstanding role in telecom chain. By analyzing the relation among the players, a dynamic relation model between carrier and service provider was provided to solve this tri-game problem and its solutions were found. Furthermore, the stability of the solution was also investigated via isoclinic lines method. In addition, supposing the game model was disturbed because of carrier's unexpected price adjustment, three responses of service provider were discussed with corresponding comments. By quantitative analysis, finally the work made a conclusion that, in order to keep a sustainable development in an uncertain market, service provider must endeavor to enhance the service's attraction by constantly improving quality of its content.

Keywords: Pricing strategy, Telecom industry, Isoclinic lines method, Tri-game.

1 Introduction

Along with the coming of the third generation mobile communication (3G) age, the carrier status in telecom chain has a subtle change. In china, carrier used to be a dominator in the industry chain, but in the future whether he will continue to maintain the leading position is depended on his business strategy. There is a dynamic relation among users, carrier and his downstream's content provides (CP). So the adjustment of the carrier strategy, especially pricing strategy, will directly affect the development of the entire industry chain. There are a lot of references that talk about this problem by qualitative method, but few use the quantitative analysis to study it. This is a tri-game problem. To solve this kind of model, Nash has provided a methodology of agent analysis, which makes the complicated interrelations easy^[1-5]. Based on it, this article provides a dynamic relation model to discuss the game players' pricing strategy and the stability of the result.

The remaining section of the article is organized as follows: the relation model is described in Sect.2. The stability of the solution is discussed in Sect. 3. The anti-impact ability of CP is analyzed in Sect.4. Finally, the conclusions are given in Sect.5.

2 Relation Model

When users surfing the internet, they should not only pay the network working cost to the carrier, but also provide the corresponding content working cost to CP [6-8]. So a dynamic relation process is formed between carrier and CP by users. The adjustment of network flow fee will directly affect the surfing frequency and time, and then affect the income of CP. However, the price adjustment of one content provide will only affect the profitability of himself, but will not involve the whole industry. Accordingly, it will not affect the operator's overall revenue.

The network flow is impacted by the current network flow fee. Carrier is willing to provide the capacity of the network associated with the expected proceeds, and the expected proceeds are influenced by the expected price. In this article, we assume that the expected price is equal to the initial price. Motivated by this assumption, the relation between carrier and users is defined as

$$\begin{cases} d_1(t) = a - bp_1(t) \\ s_1(t) = kp_1(t-1) - l \end{cases} \quad (1)$$

Where, $d_1(t)$, $s_1(t)$ represents the network flow and the network capacity, and $p_1(t)$ is the network flow fee. a, b, k, l are the parameters of this system.

The relation between CP and users is defined as

$$\begin{cases} d_2(t) = m - n[p_1(t) + p_2(t)] \\ s_2(t) = xp_2(t-1) - yp_1(t-1) - z \end{cases} \quad (2)$$

Where, $d_2(t)$, $s_2(t)$ represents the content flow and the content capacity, and $p_2(t)$ is the price of the content. m, n, x, y, z are the parameters of this system.

When the systems of both carrier and CP have reached the stability, they become

$$\begin{cases} q_1(t) = d_1(t) = s_1(t) \\ q_2(t) = d_2(t) = s_2(t) \end{cases} \quad (3)$$

So, we obtain

$$\begin{cases} p_1(t) = \frac{a+l}{b} - \frac{k}{b} p_1(t-1) \\ p_2(t) = \frac{m+z}{n} + \frac{y}{n} p_1(t-1) - p_1(t) - \frac{x}{n} p_2(t-1) \end{cases} \quad (4)$$

When the two markets are both in an equilibrium status, the prices satisfy

$$\begin{cases} p_1^* = p_1(t) = p_1(t-1) \\ p_2^* = p_2(t) = p_2(t-1) \end{cases} \quad (5)$$

From Eq. (4) and Eq. (5), the equilibrium price can be obtained as

$$\begin{cases} p_1^* = \frac{a+l}{b+k} \\ p_2^* = \frac{m+z}{n+x} + \frac{(y-n)(a+l)}{b+k} \end{cases} \quad (6)$$

3 Stability of the Model

From Eq. (4), the unit time price ratios of network and content are

$$\begin{cases} \Delta p_1(t) = p_1(t+1) - p_1(t) = \frac{a+l}{b} - \frac{k+b}{b} p_1(t) \\ \Delta p_2(t) = p_2(t+1) - p_2(t) \\ \quad = \frac{m+z}{n} - \frac{a+l}{b} + \left(\frac{y}{n} + \frac{k}{b} \right) p_1(t) - \frac{x+n}{n} p_2(t) \end{cases} \quad (7)$$

If only consider the carrier system, when $\Delta p_1(t) = 0$, the market reach, that is

$$\Delta p_1(t) = \frac{a+l}{b} - \frac{k+b}{b} p_1(t) = 0 \quad (8)$$

At this time, the CP market may not be balanced. From Eq. (8), the equilibrium price of carrier is only influenced by him, and has nothing to do with CP. This result fully shows that carrier's price will not be affected by CP. Eq. (8) expresses the equilibrium condition of the carrier market, which is denoted as

$$p_1(t) = \frac{a+l}{k+b} \quad (9)$$

The line plotted in Fig. 1 is called isoclinical lines. But it is a special isoclinical line. This line has nothing to do with longitudinal axis, which deputizes for the CP's market. That is to say, this line is perpendicular to the horizontal axis.

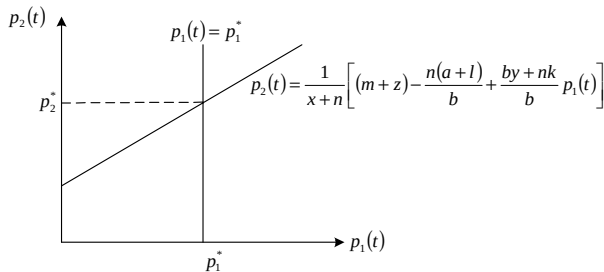


Fig. 1. The Phase plane of carrier and CP

If only consider CP system, in accordance with the same method we can get the isoclinical line of CP market, which is expressed as

$$\Delta p_2(t) = \frac{m+z}{n} - \frac{a+l}{b} + \left(\frac{y}{n} + \frac{k}{b} \right) p_1(t) - \frac{x+n}{n} p_2(t) = 0 \quad (10)$$

From Eq. (10), there is

$$p_2(t) = \frac{1}{x+n} \left[(m+z) - \frac{n(a+l)}{b} + \frac{by+nk}{b} p_1(t) \right] \quad (11)$$

This line expresses the relation between $p_1(t)$ and $p_2(t)$ when CP market reaches equilibrium.

Fig.1 shows the phase plane of carrier and CP. There is a designated market in equilibrium status when the prices change along the corresponding line. The intersection of two lines which is denoted as (p_1^*, p_2^*) means the two markets are both in equilibrium. When the prices are not in these lines, i.e. in his top/ below or left/ right, the markets can not be in equilibrium status. These lines divide the plane into four quadrants named as I、II、III、IV. In each quadrant, respectively choose four points a、b、c、d shown in Fig. 2.

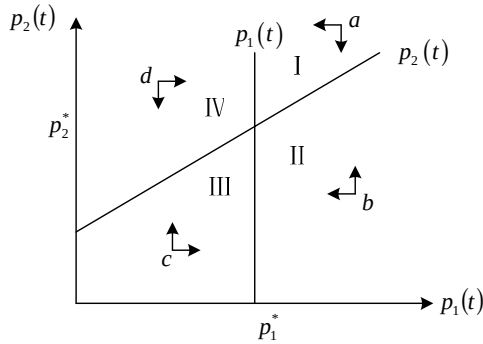


Fig. 2. Four quadrants in the phase plane

If only consider the carrier system, when $\Delta p_1(t) > 0$, there is

$$p_1(t) < \frac{a+l}{k+b} = p_1^* \quad (12)$$

That is to say, when the price decided by carrier is less than the equilibrium price, $p_1(t)$ will increase. This result is shown in Fig. 2 through the horizontal arrows from c and d which point right. Similarly, when $\Delta p_1(t) > 0$, there is

$$p_1(t) > \frac{a+l}{k+b} = p_1^* \quad (13)$$

It means when the price decided by carrier is higher than the equilibrium price, $p_1(t)$ will decrease. This result is shown in Fig. 2 through the horizontal arrows from a and b which point left.

If only consider CP system, when $\Delta p_2(t) > 0$, there is

$$p_2(t) < \frac{1}{x+n} \left[(m+z) - \frac{n(a+l)}{b} + \frac{by+nk}{b} p_1(t) \right] \quad (14)$$

That is to say, when the price decided by CP is less than the equilibrium price, $p_2(t)$ will increase. This result is shown in Fig. 2 through the horizontal arrows from b and c which point top. Similarly, when $\Delta p_2(t) < 0$, there is

$$p_2(t) > \frac{1}{x+n} \left[(m+z) - \frac{n(a+l)}{b} + \frac{by+nk}{b} p_1(t) \right] \quad (15)$$

It means when the price decided by CP is higher than the equilibrium price, $p_2(t)$ will decrease. This result is shown in Fig. 2 through the horizontal arrows from a and d which point below.

These vectors respectively denote the powers which put on the system from two directions. The prices in the four quadrants are all controlled by the powers, which are shown in Fig. 2. The powers force the system to tilt the direction of the intersection. So we can get the solution that the equilibrium result is steady.

4 The Anti-impact Ability of CP

With the ever-aggravating competition, the ever-developing technology and the costumers' ever-changing demands, carrier may adjust prices to meet the market requirements. When carrier suddenly adjusts prices, the two systems will has a disturbance. At this time, how to do is the problem that each CP must face.

From Eq. (11), if carrier enhances the price, that is $p_1'(t) > p_1^*$, CP must increase the content price to get the equilibrium solution. Based on Eq. (12), when $p_1'(t)$, $p_2'(t)$ are both higher than ever, the marker requirement of this content must decrease. The increased demand to internet business meets the spread of the virus effect, while the decreased demand also meets this characteristic. So CP must prevent the demand decline.

In order to retain customers, according to Eq. (12) CP can take the following three methods:

- Reduce their prices $p_2'(t)$

When $p_1'(t)$ increase and $p_2'(t)$ decrease, CP's profit margins will inevitably fall sharply. So The way to reduce their prices is not a good strategy.

- Enhance the parameter m

The parameter m is related to the largest market demand. However the largest demand is very difficult to elevate, that is to say the parameter m is not easy to change. So this is a difficult executive strategy.

- Reduce the parameter n

The parameter n expresses the influence level of the price to users. The bigger is n , the greater is the impact on users. Market demand to an internet business is depended on not only the price but also the quality of it^[9-10]. When the content sufficiently attracts the attention of users, the user's reliance level on the price will reduce. So if CP wants to reduce the parameter n , he must ever-improve the quality of his business. In this condition, when carrier increases the price of his business, users will not instantly loss. This is a good strategy to solve this problem.

5 Conclusions

This article uses a dynamic relation model to analyze the tri-game players' pricing strategy in the telecom industry. By solving this model, the equilibrium solution is provided, and the solution's stability is also discussed via isoclinic lines method. The investigation shows that the solution is a Nash equilibrium point. Furthermore, supposing carrier suddenly adjusts his pricing strategy, which makes the system disturb, CP has three strategies to withstand shocks. But only one way that is ever-improve the quality of his content can help CP maintain his users so that he can make sustainable development in the ever-changing competition market. The conclusion of this article can provide a reference for carrier and CP when they consider how to make the pricing strategy.

References

1. Stoft, S.: Using game theory to study market power in simple networks. In: Singh, H. (ed.) IEEE Power Eng. Soc. Winter Meeting, New York, pp. 33–40 (1999)
2. Fudenberg, D., Tirole, J.: Game theory, 5th edn. MIT Press, Cambridge (1996)
3. Nobel laureates's perspectives on China's economy, pp. 289–310. Economy Daily Press, Beijing (2006)
4. Li, J., Das, T.K.: Learning Nash equilibrium policies in average reward irreducible stochastic games. Working paper, Dept. of Industrial and Management Systems Engineering, Univ. South Florida, Tampa, FL (2003)
5. Lin, J.: A labor market game between enterprises and ELISA. Journal of Chongqing University of posts and telecommunications 04, 87–90 (2002)
6. Stanley, B., Paul, E.F., Madhav, V.R.: Performance measurement and design in supply chain. [J]. Management Science 47(1), 173–188 (2001)
7. Janjua, K.A., Khan, S.A., Noor, S.A.: Market analysis of competition in US local telephony. In: International Conference on Wireless Communications, Networking and Mobile Computing, WiCOM 2007, pp. 4970–4973 (2007)
8. Cunningham, L.B., Baldick, R., Baughman, M.L.: An empirical study of applied game theory: transmission constrained Cournot behavior. IEEE Trans. Power Syst. 17, 166–172 (2002)

9. Tsai, H.-C., Chen, C.-M., Tzeng, G.-H.: The comparative productivity efficiency for global telecoms. *International Journal of Production Economics* 103(2), 509–526 (2006)
10. Ruo-peng, L., Ting-jie, L.: Data service: characteristics, market structure, and the role of operator. *The Journal of China Universities of Posts and Telecommunications* 12(2), 35–39 (2005)

A Semantic Description Approach for Telecommunications Network Capability Services

Xiuquan Qiao, Xiaofeng Li, and Tian You

State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications, Beijing, 100876, China
{qiaoxq,xfli}@bupt.edu.cn, youtian1985@gmail.com

Abstract. The provision of pervasive services in the ubiquitous convergent network presents a great challenge in light of well known problems like context-awareness, semantic service description, accurate service query and dynamic service composition. In order to eliminate the semantic gap of Telecommunication Network and Internet in the service layer, we propose the provision of the semantic Telecommunications Network Capability Services (TNCS). By comparing the differences between TNCS and the plain Web Services, we present a semantic description method of TNCS by extending OWL-S. Then using this approach, we demonstrate a service profile description case for semantic ParlayX. In this way, the ontology-based accurate query, matching, automatic composition and invocation for telecommunication network capability services can be supported. The semantic convergence of telecommunication network and Internet in the service layer could be facilitated.

Keywords: Semantic Parlay X, Semantic Parlay, Telecommunication Network Capability Service, Telecommunication Service Domain Ontology, Service-Oriented Architecture, Ontology, Semantic Web Service, OWL-S.

1 Introduction

The future ubiquitous convergent network (B3G/4G) is one user-centric harmony communication network. It will integrate various heterogeneous networks (i.e. mobile/fixed network, Internet and some emerging networks) and new technologies. The target of ubiquitous convergent network is to provide user-centric pervasive personalization service. It can freely enable the information exchange between people and people, people and entity, as well as entity and entity at anytime, anywhere, and anyway. Therefore, the new ubiquitous convergent services need transparently adapt to all kinds of terminals, access networks. Based on this infrastructure, the future services would be characterized by pervasive access, context-awareness and seamless mobility. As the goal of user-centric service is to facilitate the work and daily life of human, it will involve with various domain services, such as telecom domain service, financial service, transportation service. Service will be more intelligent. This requires that intelligent service agent could accurately discover, automatically compose and invoke the services provided by telecommunication network or Internet based on the service context in the service-oriented architecture.

In order to support the accurate service discovery, the Telecommunication Network Capability Services (TNCS) like call control and message sending/receiving must be described in the semantic level. However, the open interface specifications of telecommunication network are still in the syntactic description level. Currently, NGN and 3G network all adopt the open API (Application Programming Interface) technologies in the service layer, such as Parlay/OSA [1], ParlayX [2]. The TNCS, such as call control, conference management, SMS sending/receiving and location, are available to the service developers in the form of API. This greatly facilitates the value-added service development. Moreover, the Service-Oriented Architecture (SOA) is imported into the telecom service domain by Parlay Web Service specifications. However, WSDL-based open interface specifications lack a semantic description for TNCS. This kind of interface is faced to service developers, not to machine. It is very difficult for service agent to achieve the accurate discovery and automatic invocation of TNCS. So the current TNCS interface specifications based on WSDL are unable to support the provision of advanced intelligent ubiquitous service.

Today, semantic web service, as a new research paradigm, is generally defined as the augmentation of web service description through semantic annotation, to facilitate the higher automation of service discovery, composition, invocation and monitoring in an open environment. As a submitted standard of W3C, OWL-S [3] is OWL [4] - based ontology description framework for web service. However, OWL-S can not be directly used to describe TNCS. The reason is: On the one hand, OWL-S, as a common semantic web service description framework, is unable to perfectly describe some important features of TNCS, such as charging, network or terminal characteristics; On the other hand, the semantic description of TNCS needs the telecommunication service domain ontology to support ontology-based reasoning and knowledge sharing. But OWL-S does not involve the definition of domain ontology.

In this article, we import the semantic web service and ontology technologies into the telecommunication service domain, and present TelecomOWL-S for the semantic description of TNCS by extending OWL-S. Moreover, telecommunication service domain ontology is proposed to support the nonfunctional properties description for TNCS. It establishes the foundation of ontology-based reasoning and domain knowledge sharing. The proposed approach enhances the accuracy of telecommunications network capability service description and matching, and eliminates the semantic gap between telecommunications network and Internet in the service layer. The rest of this paper is structured as follows. In section 2 we introduce the relevant prior work. Section 3 we present the complete semantic description approach for telecommunications network capability services. In section 4, conclusions are then drawn.

2 Related Work

The integration of Semantic Web Service (SWS) technology and telecommunications systems is currently the subject of intensive research. In this section, we provide an overview of the related work. So far, a number of researches have been done in applying ontology and semantic web service technologies to mobile service domain.

To solve the lack of QoS/QoE considerations of the existing Parlay X, Sungjune Hong [5], [6] presented the extended Parlay X with QoS/QoE for 4G network. In fact, this paper imported the context-awareness technology into the Parlay X gateway. The

proposed Parlay X gateway can provide newly defined API such as `getContextAwareness` and `getPersonalization`. Songtao Lin [7] suggested that roaming users should use local service as possible by semantic-based service discovery, composition and invocation. However, this paper mainly focused on the provision VHE service by the Intelligent Mobile Service Platform based on semantic web technologies, not the provision of open semantic TNCS. Tomas Vitvar [8] discussed how semantic web services technology can facilitate dynamic and optimal integration of voice and data services with specific characteristics conforming users needs and preferences. Villalonga, C. [9] provided an overview of the Mobile Ontology based on the need for a standardized ontology that describes semantic models of the domains relevant for scalable NGN service delivery platforms. This work, as a part of IST SPICE project, is a meaningful attempt to establish a standardized ontology for mobile service delivery in NGN. However, this mobile ontology is used to address the semantic sharing among the distributed components of SPICE service platform. So the mobile ontology has its limitation. Moreover, the mobile ontology is still in an early phase and will need to be extended. Additionally, Alistair Duke [10] explored the use of semantic web service within the Operational Support System (OSS). This paper discussed the dynamic B2B integration with semantic web service technologies. However, this approach only focuses on the telecommunications management.

From the above analysis, we can see that the existing work has not explored the semantic description and provision problem of TNCS with the semantic web services.

3 Semantic Description Approach for Telecommunications Network Capability Services (TNCS)

3.1 Some Differences between TNCS and the Plain Web Services

Compared with the plain web services, the TNCS, especially that provided by mobile network, have some distinct domain-related characteristics.

From the terminals aspect, the TNCS provided by different networks maybe have different terminal requirements. The provision of telecommunication service is greatly dependent on the capabilities of terminals. As communication network technologies are evolving constantly, there are great capability differences among the different terminals, such as terminal browser, communication protocol. For example, the location service provided by CDMA network requires that the terminals support the GPSOne positioning technology. However, the location service provided by GSM network based on Cell-ID positioning technology almost has no special requirements for terminals. In order to deliver the seamless service on the user's different terminals, the service delivery platform must discover the appropriate TNCS based on the service context. So the semantic description of TNCS should consider the requirements on terminals.

From the network aspect, currently, there are a variety of communication networks, such as GSM, CDMA, fixed network, WLAN. The different networks have the different service quality. For example, the wireless networks are less reliable than line-based networks. To meet the different user's needs, service delivery platform should select the appropriate TNCS based on the Service Level Agreement (SLA). So the non-functional features like network type, service bandwidth, and communication mode are important for the semantic description of TNCS.

In addition, a significant difference between telecom service and Internet service is the charging pattern. Internet services are often free, however telecom services have various charging model, such as event-based, session-based, time, volume. In order to satisfy the different consuming levels of users, the intelligent agent or service delivery platform should query the suitable TNCS based on the user's cost preference. So, the charging model should be the important part of the semantic description of TNCS.

From the above analysis, we can see that the provision of TNCS is greatly constrained by the network condition, terminal capability and other non-functional features. The TNCS provided by a concrete network has its own using scope and conditions. Therefore, in order to provide the personalized services, the intelligent service platform should dynamically select the appropriate TNCS or other value-added services based on the service context info. To achieve this goal, the TNCS need to be accurately described in the semantic level so that the annotated semantic information can fully reflect its characteristics.

3.2 The Involved Problems of the Semantic Description of TNCS

In order to enable the service agent to accurately discover and automatically invoke the TNCS based on the service context, TNCS should be described in the semantic level, not the syntactic level. This work needs to solve two issues:

- (1) **A Tailored Semantic Description Ontology for TNCS:** TNCS, especially that provided by mobile network, has some special non-functional features, such as network characteristics, billing policy, terminal capability requirements, and quality of service. These non-functional features are very important for context awareness service. Currently, OWL-S provides a high-level ontology description framework for general semantic web service. However, OWL-S is unable to fully support the important characteristics description of TNCS. In order to fully reflect the important features of TNCS, a tailored semantic description ontology for TNCS is needed. In this paper, we present TelecomOWL-S, which is an extension of OWL-S, to solve this issue, see section 3.3.
- (2) **The Ontology-based Formal Description of TNCS-related Domain Concepts and Knowledge:** When semantically describing TNCS, its input/output parameters and some important service features like network type and terminal requirements involve with a lot of domain concepts and knowledge. In order to support the ontology-based reasoning and knowledge sharing, these concepts must be modeled in the formal way. However, telecommunications service domain includes a large number of concepts and terminology. How to organize these concepts and finally form a knowledge system is a great challenge. In this paper, we present Telecommunication Service Domain Ontology (TSDO) and TNCS-related Application Ontology to solve this issue, see section 3.4. and section 3.5.

3.3 TelecomOWL-S: An Extension of OWL-S

The semantic description of TNCS is the formal specification of the open interfaces provided by the telecommunications network. The concrete logic implementation of TNCS is not involved. So the ServiceModel and ServiceGrounding of OWL-S are sufficient for the semantic description of TNCS. Specifically, the atomic process

ontology of ServiceModel can be directly used to model the network operations such as MakeACall, CancelCall. The ServiceGrounding ontology can be used to mapping the atomic process of ServiceModel to the operation of WSDL-based open interface specification like Parlay X. Therefore, the ServiceProfile of OWL-S needs to be extended to describe the related important features of TNCS.

In fact, the service profile is used to characterize a service for purposes such as advertisement, discovery, and selection. Service profiles may be published in various kinds of registries, discovered using various tools, and selected using various kinds of matchmaking techniques. OWL-S does not prescribe or limit the ways in which profiles may be used, but rather, seeks to provide a basis for their construction that is flexible enough to accommodate many different contexts and methods of use. In order to accurately describe the important features of TNCS, we present a tailored service ontology--TelecomOWL-S, which mainly extends the ServiceProfile ontology of OWL-S.

Similar to OWL-S, TelecomOWL-S also has three parts: TelecomServiceProfile, ServiceModel, and ServiceGrounding. As the semantic description of TNCS does not involve the complex service logic control, TelecomOWL-S adopts the same ServiceModel and ServiceGrounding ontology as those of OWL-S. The difference is that TelecomOWL-S redefines a new TelecomServiceProfile by extending ServiceProfile of OWL-S based on the characteristics of TNCS. Fig.1. depicts the high-level ontology overview of TelecomOWL-S.

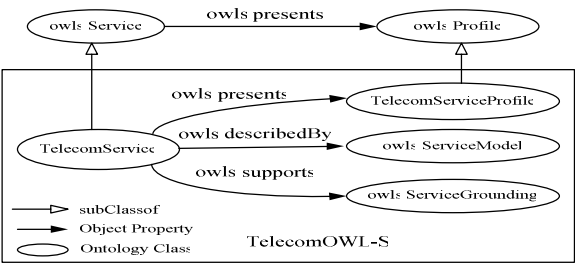


Fig. 1. The High-level Ontology of TelecomOWL-S

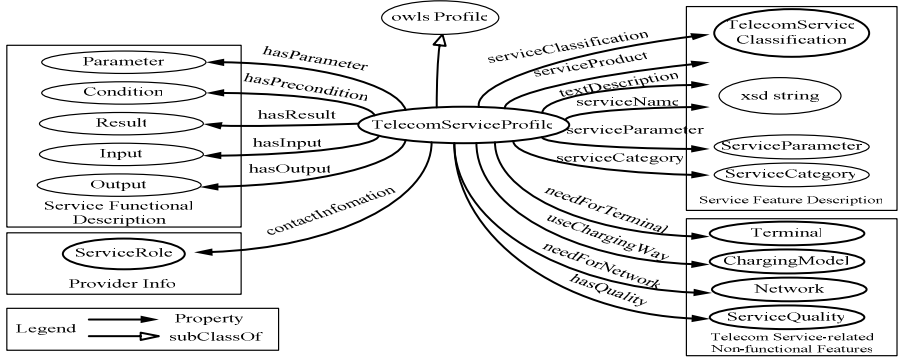


Fig. 2. TelecomServiceProfile Ontology

The TelecomServiceProfile is directly derived from ServiceProfile of OWL-S. The extension ways mainly include two kinds, one is to limit the range of the existing property, and the other is to define new special property. Specifically, the classes and properties of TelecomServiceProfile ontology are shown in Fig.2.

The TelecomServiceProfile mainly consists of 4 parts: Service Functional Description, Service Provider Information, Service Feature Description and Telecom Service-related Non-functional Feature. The former three parts mainly inherits from OWL-S. Only the value ranges of some properties like serviceClassification, contactInformation are constrained by the specific concepts defined in the TSDO. The last part is mainly used to describe some peculiar features of TNCS. Currently, four new object properties are defined.

- (1) **needForTerminal**: This property can be used to depict the service requirements for user’s terminal capabilities, such as terminal browser, terminal hardware, terminal software and WAP. The concepts and terminology about terminal capability are from the Terminal Capability Ontology of TSDO.
- (2) **useChargingWay**: This property is defined to describe the service-related various billing policies and corresponding tariffs, such as time-based, volume-based, event-based, and flat fee. The concepts and terminology about charging are from the Charging Ontology of TSDO.
- (3) **needForNetwork**: The characteristics of network providing service can be described by this property, such as the network type, network bandwidth. The concepts and terminology about charging are from the Network Ontology of TSDO.
- (4) **hasQuality**: This property can describe the service quality, such as response time, connectivity, delay. The relevant concepts and terminology are from the Service Quality Ontology of TSDO.

3.4 Telecommunication Service Domain Ontology (TSDO)

The semantic descriptions of TNCS need the support of telecommunications service domain concepts and knowledge. Telecommunication Service Domain Ontology (TSDO) provides some shared domain concepts and knowledge about telecom service. This will facilitate the semantic sharing and interoperability of heterogeneous communication entities.

Considering the scalability and flexibility of TSDO, we construct the domain ontology in a hierarchical structure. TSDO only provides some core domain concepts. Based on these core domain ontologies, we can construct various concrete

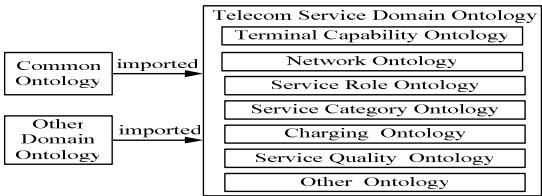


Fig. 3. Telecommunication Service Domain Ontology

application-related ontologies, such as TNCS-related application ontology or service context ontology. Based on a modular design principle, it mainly comprises six sub-ontologies. Fig.3. shows the overview of TSDO. In addition, the construction of TSDO needs the support of some Common Ontology and other domain ontology, such as time ontology, location ontology.

- (1) **Terminal Capability Ontology:** defines some main concepts about terminal software, terminal hardware, terminal browser and network characteristics supported by terminal. Currently, CC/PP and UAProf, which are defined by RDF (Resource Description Framework) language, are used to describe the terminal capability. They cannot be directly used to describe the semantic TNCS. So we create the Terminal Capability Ontology by OWL language based on CC/PP and UAProf specifications.
- (2) **Network Ontology:** mainly specifies the network concepts, network category, network features, as well as the relationships of various networks, such as mobile network, internet, and fixed network, GSM, CDMA, UMTS, WCDMA, WLAN.
- (3) **Service Role Ontology:** describes the stakeholders' concepts of the service supply chain, for example, service provider, content provider, network operator, service user.
- (4) **Service Category Ontology:** descriptions of telecommunication service classification. As a proposed service category standard by OWL-S, the UNSPSC (United Nations Standard Products and Services Code) provides an open, global multi-sector standard for efficient, accurate classification of products and services. It often used in the E-commerce field. However, on the one hand, UNSPSC is not based on ontology, so it is only suitable for the serviceCategory property of ServiceProfile, not the serviceClassification property of ServiceProfile. On the other hand, UNSPSC has no concrete telecommunications service classification, only to Telecommunication Services (code: 81161700) level. Therefore, UNSPSC has no ability to enable the accurate telecommunications service query. So we construct the Service Category Ontology in TSDO to enable the accurate description of TNCS. It defines the relationship between various telecommunications services, like basic service, value-added service, voice service, data service, conference service, telecommunication network capability service, download service, browsing service, messaging service.
- (5) **Charging Ontology:** defines the charging-related concepts about telecommunication service, includes payment way (such as prepaid and postpaid), charging type (such as time-based, volume-based, event-based, content-based), billing rate, as well as account balance.
- (6) **Service Quality Ontology:** Telecommunication network must provide the services which have the end-to-end QoS guarantee. As the technical differences, the quality of service provided by different networks is different. Service Quality Ontology mainly defines the QoS-related concepts about telecommunication service, including access network QoS, core network QoS and user's QoE, such as call delay, message size, call through rate, positioning accuracy, network bandwidth.

As the length limit of this article, a part of Network Ontology is shown in Fig.4.

```
<owl Class rdfID="Network"/>
<owl Class rdf about="#TelecomNetwork">
  <rdfs subClassOf rdf resource="#Network"/>
</owl Class>
<owl Class rdf about="#MobileNetwork">
  <rdfs subClassOf>
    <owl Class rdf about="#TelecomNetwork"/>
  </rdfs subClassOf>
  <owl disjointWith rdf resource="#FixedNetwork"/>
</owl Class>
<owl Class rdfID="GSM">
  <rdfs subClassOf rdf resource="#MobileNetwork"/>
</owl Class>
<owl ObjectProperty rdfID="operatedBy">
  <rdfs range rdf resource="#ServiceRole;#NetworkOperator"/>
  <rdfs domain rdf resource="#TelecomNetwork"/>
</owl ObjectProperty>
```

Fig. 4. A part of Network Ontology

3.5 The Application Ontology of TNCS

The different TNCS has the different interface parameters, i.e. input/output. In order to describe the semantic TNCS, theses interface parameters must be described in the ontology format. Therefore, the related application ontology needs to be created, such as Parlay X Application Ontology or Parlay Application Ontology, see Fig.5.

The concrete application ontology modeling mainly referenced of the existing XML-based parameter type definition of Parlay X or Parlay specifications. The main

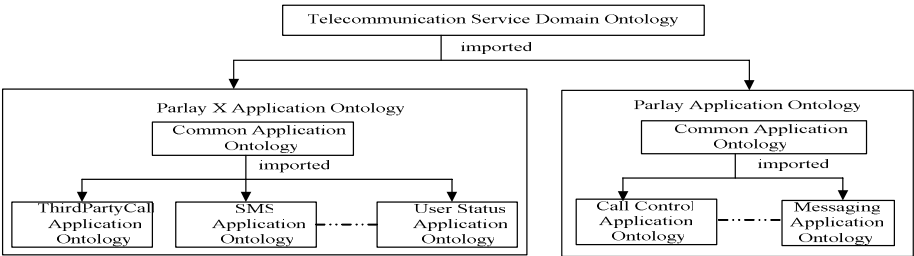


Fig. 5. The Application Ontology Overview of TNCS

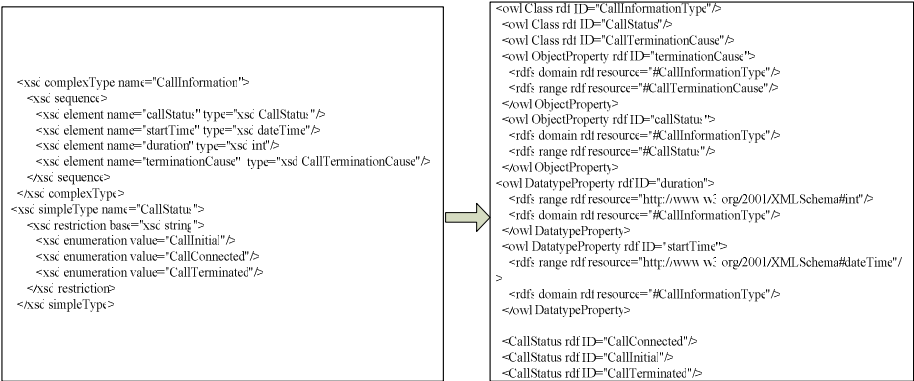


Fig. 6. Part of ThirdPartyCall Application Ontology for Semantic ParlayX

advantage is that ontology-based input/output can enable the reasoning and accurate matching. Fig.6. shows a comparison of XML-based parameter description and ontology-based parameter description in the ThirdPartyCall application ontology for semantic ParlayX service.

3.6 The Description Case of Semantic Parlay X Service

Using the TelecomOWL-S, TSDO and the concrete application ontology, we can describe the semantic TNCS with Protégé tool. As the modeling method of ServiceModel and ServiceGrounding is similar to the plain semantic web service, in this paper we mainly focus on the description of TelecomServiceProfile, and demonstrate how to combine the TelecomServiceProfile, TSDO and application ontology. Fig.7. shows a part of ThirdPartyCall Service Profile of Semantic ParlayX service. As the length limit of this article, the extended properties which depict the service quality, network type, charging way are shown. Fig.7. clearly depicts the important features of ThirdPartyCall service. Specifically, it belongs to the ThirdPartyCall service in the

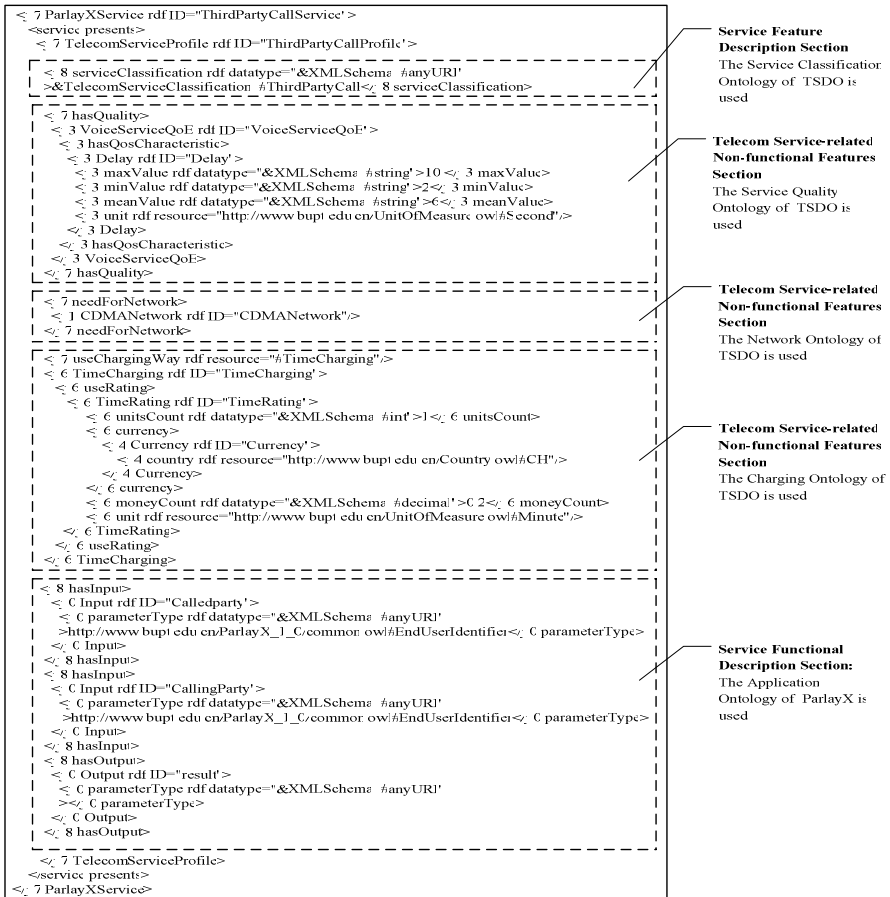


Fig. 7. A Part of ThirdPartyCall Service Profile of Semantic ParlayX

telecommunication services classification. It is provided by CDMA network. In the service quality aspect, its maximum delay time: 10 seconds, minimum delay time: 2 seconds, mean delay time: 6 seconds. The time-based charging way is used, and the charging rate is 0.2 RMB per minute.

4 Conclusions

In this paper we presented a semantic description approach for telecommunication network capability services. Like the existing ParlayX or Parlay gateway, the new semantic ParlayX/Parlay gateway can provide the semantic TNCS. This will greatly eliminate the semantic gap between telecom network and Internet in the service layer. So far, we have implemented a prototype system, including semantic Parlay X gateway, telecom service domain ontology repository, Parlay X application ontology, semantic UDDI, intelligent agent. The cheapest third party call service is developed.

However, the research of semantic TNCS is still in an early phase. The TSDO need to be further enhanced. The standardization of semantic TNCS is still a long way. Currently, we are cooperating with Huawei Corp. to promote this work.

Acknowledgments. This work is supported by National Natural Science Foundation of China project “Service Intelligence for Convergent Network Environment” (No. 60672122) and Specialized Research Fund for the Doctoral Program of Higher Education (No. 20070013026).

References

1. Parlay Group, Parlay API 4.0, <http://www.parlay.org>
2. Parlay x, <http://www.parlay.org/en/specifications/pxws.asp>
3. OWL-S, <http://www.w3.org/Submission/2004/07/>
4. Web Ontology Language (OWL), <http://www.w3.org/2004/OWL>
5. Hong, S.J., Han, S.Y., Choi, B., Kim, Y.J., Ahn, C.H.: The semantic PARLAY for 4G network. In: 2nd International Conference on Mobile Technology, Applications and Systems, pp. 15–17 (2005)
6. Hong, S.J., Han, S.Y., Song, K.: The Extended PARLAY X for an Adaptive Context-Aware Personalized Service in a Ubiquitous Computing Environment. In: Enokido, T., Yan, L., Xiao, B., Kim, D.Y., Dai, Y.-S., Yang, L.T. (eds.) EUC-WS 2005. LNCS, vol. 3823, pp. 288–297. Springer, Heidelberg (2005)
7. Lin, S.T., Chen, J.L.: Semantic Web Enabled VHE for 3rd Generation Telecommunications. In: 4th Annual ACIS International Conference on Computer and Information Science (2005)
8. Vitvar, T., Viskova, J.: Semantic-enabled Integration of Voice and Data Services: Telecommunication Use Case. In: Third IEEE European Conference on Web Services (ECOWS) (2005)
9. Villalonga, C., Strohbach, M., Snoeck, N., Sutterer, M., Belaunde, M., Kovacs, E., Zhdanova, A.V., Goix, L.W., Droegehorn, O.: Mobile Ontology: Towards a Standardized Semantic Model for the Mobile Domain. In: Proceedings of the 1st International Workshop on Telecom Service Oriented Architectures (TSOA 2007) at the 5th International Conference on Service-Oriented Computing, Vienna, Austria, 17 September (2007)
10. Duke, A., Richardson, M., Watkins, S., Roberts, M.: Towards B2B Integration in Telecommunications with Semantic Web Services. In: Gómez-Pérez, A., Euzenat, J. (eds.) ESWC 2005. LNCS, vol. 3532, pp. 710–724. Springer, Heidelberg (2005)

Low Latency Proactive Handover Scheme for Proxy MIPv6 with MIH

Igor Kim, Young Chul Jung, and Young-Tak Kim*

Dept. of Information and Communication Engineering, Graduate School,
Yeungnam University. 214-1, Dae-Dong, Gyeongsan-si, Gyeongsanbuk-do, 712-749, Korea
ikim@ynu.ac.kr, callaguy@yumail.ac.kr, ytkim@yu.ac.kr

Abstract. This paper proposes a novel low latency handover scheme for Proxy Mobile IPv6 (PMIPv6) using Media Independent Handover (MIH) services. We analyze the major aspects that induce high latency during Mobile Node's (MN) handover within a PMIPv6 domain. Our analysis consists of Layer 2 (channel scanning, MN authentication, MN's profile acquisition) and Layer 3 (binding registrations and tunnel establishment) operations. By using MIH services we provide a solution that minimizes overall scanning delay at Layer 2, the information service provided in MIH can provide context transfers, and efficient triggering events can speed up handover procedure. Through the performance evaluations made in the OPNET network simulator we validate our proposed scheme.

Keywords: PMIPv6, MIH, MAG, LMA, QoS, PBU, PBA.

1 Introduction

During the last decade the usage and importance of wireless networks significantly increased. Nowadays, mobile devices equipped with wireless interface(s) are widely spread in the market and used by people in their daily life. IETF standardized the protocols that can support service continuity when user changes its location. Mobile IPv6 [1] has been proposed by IETF to support mobility in IPv6 networks. However, it has high handover latency that is not acceptable for real-time multimedia applications. For this reason, a MIPSHOP working group of IETF developed a series of standards, such as Fast handovers for mobile IPv6 (FMIPv6) [2] and Hierarchical mobile IPv6 (HMIPv6) [3] directed to solve handover delay latency and reduce the signaling overhead. However, all of this global mobility protocols require a Mobile Node (MN) to support Mobile IPv6 in their protocol stack. This can generate a number of problems, since there are protocol stacks that do not support Mobile IPv6. NetLMM working group started its work to provide network-based mobility management. The new approach is called Proxy Mobile IPv6 (PMIPv6) [4] and supports mobility for IPv6 nodes by extending Mobile IPv6 signaling and reusing the Home Agent (HA) via a proxy mobility agent in the network. Supporting mobility does not require a mobile

* Corresponding author.

node to be involved in the signaling required for handover process. The proxy mobility agent performs mobility management on behalf of the mobile node.

Currently, the PMIPv6 basic specification describes procedures that should be made during the initial attachment of the MN to the PMIPv6 domain and handover signaling message flow when mobile node performs handover within a PMIPv6 domain. However, handover latency in PMIPv6 negatively affects the QoS of real-time applications such as VoIP and videoconference. Typically, it is assumed that handover latency should not be more than 50 ms in order to provide seamless service. In PMIPv6 it is very essential to recognize MN's attachment and detachment events. As soon as a Mobile Access Gateway (MAG) recognizes a MN's attachment/detachment, it should perform registration/deregistration with Local Mobility Anchor (LMA). However, not only L3 operations must be considered, but also L2 operations must be considered impact overall handover delay. One of the most important parts of the handover at L2 is scanning, that introduces very long delay. Thus, a novel approach is needed to provide an optimization during MN's handover within PMIPv6 domain.

IEEE 802.21 working is developing a standard called Media Independent Handover (MIH) [5] which defines extensible media access independent mechanisms that enable the optimization of handovers between heterogeneous IEEE 802 systems.

In this paper, we provide analysis of handover latency introduced by PMIPv6 and propose a solution to provide seamless handover using MIH. By using MIH services we provide a solution that reduces overall scanning time at Layer 2, the information service provided in MIH can provide context transfers, and efficient triggering events can minimize handover time. In addition, a predictive handover for PMIPv6 based on FMIPv6 procedures is proposed. In this approach, a target network aware about the MN's pending registration prior to its actual attachment. Through the performance evaluations made in network simulator OPNET, we analyze handover latency and compare our proposed scheme with PMIPv6.

The rest of the paper is organized as follows. In section 2, PMIPv6 and MIH are explained. In section 3, the proposed enhanced handover scheme is introduced. Section 4 provides performance analysis of PMIPv6 and the proposed scheme. Finally, section 5 concludes the paper and future work is defined.

2 Related Work

2.1 Proxy Mobile IPv6 (PMIPv6)

Recently IETF Network-based Localized Mobility Management (NetLMM) Working Group has standardized a protocol to support local mobility management. This protocol is called Proxy Mobile IPv6 [4]. The major reason of the developing such protocol is that currently there exist IPv6 stacks that do not have Mobile IPv6 functionality and there would likely be IPv6 stacks without Mobile IPv6 client functionality in the future as well. It is intended for providing network-based mobility management support to a mobile node, without requiring the participation of the mobile node in any mobility related signaling [4]. In general, there are three main well-known problems involved in using a global mobility protocol for every movement between ARs [6]: update latency, signaling overhead, and location privacy.

The core components of a PMIPv6 domain are Mobile Access Gateway (MAG) and Local Mobility Anchor (LMA). The LMA is a topological anchor point for the mobile node's home network prefix and responsible for maintaining mobile node's reachability state. The purpose of the MAG is to perform mobility management on behalf of the mobile node. MAG performs movement detection of the MN and sends binding update messages to LMA.

After MN performs L2 attachment to the access network, MAG obtains MN-Identifier. This identifier must be stable across the entire PMIPv6 domain and the entities must be able to use this identifier in the signaling messages. Proxy Binding Update (PBU) message must have Network Access Identifier (NAI) option, identifying the mobile node, the Home Network Prefix (HNP) option, Timestamp option or a valid sequence number, and Link-local Address optionally [4]. Upon reception of PBU message the LMA makes a decision and generates PBA message with status field indicating the result. After MAG receives PBA message bi-directional tunnel is established between MAG and LMA. Then MN configures its address using network prefix information in the Router Advertisement message.

2.2 Media Independent Handover (MIH) Services

The purpose of the IEEE 802.21 standard is to enhance the experience of mobile users by facilitating handovers between heterogeneous networks [5]. MIH provides totally three types of service: Media Independent Event Service (MIES), Media Independent Command Service (MICS), and Media Independent Information Service (MIIS).

MIES provides event classification, event filtering, event reporting, and may be used to detect the need for handovers. It reports both local and remote events. Some of the common events that have been specified by the standard are "Link Going Down," "Link Up," "Link Down," "Link Parameters Report," and "Link Detected."

MICS enables MIH users to manage and control link behavior relevant to handovers and mobility. The commands generally carry the upper layer decisions to the lower layers on the local device entity or at the remote entity. Some examples of MICS commands are "MIH Poll," "MIH Scan," and "MIH Configure."

MIIS enables effective system access and effective handover decisions based on the information about all networks in a geographical area from any single L2 network. MIIS defines information elements and corresponding query-response mechanisms for the transfer of information. Both static and dynamic information can be provided by MIIS [5], and classified into three groups: general or access network specific information, PoA specific information, and vendor specific information.

3 PMIPv6 Handover Analysis and Low Latency Proactive Handover Scheme for PMIPv6 with MIH

3.1 PMIPv6 Handover Analysis Considering L2 Operations

PMIPv6 handover performance may vary depending on the practical approach that is applied in certain system. For instance, handover can be done with and without Duplicate Address Detection (DAD) [11], as well as solicited and unsolicited Router

Advertisement (RA) can be applied. The overall handover delay in PMIPv6 consists of the L2 and L3 processing time. We consider IEEE 802.11 access network as L2 technology. During the L2 handover sequence, MN has to perform following operations: channel scanning (T_{scan}), authentication with the target access point (T_{auth}), re-association (T_{re-ass}). During the L3 handover following procedures must be considered: MN profile acquisition time (T_{mn}), proxy binding update (PBU) and proxy binding acknowledgement (PBA) transmission and processing time (T_{PBU} , T_{PBA}), time required to perform DAD (T_{DAD}), router solicitation (RS) time (T_{RS}), router advertisement time (T_{RA}).

Handover latency at L2 can be expressed as:

$$T_{L2} = T_{scan} + T_{auth} + T_{re-ass} \quad (1)$$

The time taken for authentication depends on the type of authentication in use. In this paper, we assume that open system authentication is used. Thus, the authentication time is simply round-trip time between MN and AP (association request-reply exchange). The time needed for re-association consists of re-association request and reply message exchange between MN and AP, and additionally Inter-Access Point Protocol (IAPP) [8] negotiation between new and old AP. However, the most time consuming procedure for L2 handover is channel scanning time. Reduction of channel scanning time was already investigated in many research papers such as in [9]. As soon as MN detects link deterioration, it starts scanning on each channel finding the best channel based on the Received Signal Strength Indicator (RSSI) value. In the current 802.11 implementations handover delay may vary from hundreds of milliseconds to several seconds [9].

Handover latency at L3 can be expressed as:

$$T_{L3} = T_{MN} + T_{PBU} + T_{PBA} + T_{DAD} + T_{RS} + T_{RA} \quad (2)$$

The MN profile acquisition procedure is out of scope of the PMIPv6 draft [4]. However, there can be different methods to perform it. One of the possible solutions is to use AAA infrastructure deployed in the network. PMIPv6 domain can obtain the MN profile information during authentication phase. The time needed to perform a binding registration is simply the sum of the round trip time between MAG and LMA and the processing time. The triggering point of DAD operation depends on the underlying technology and implementation manner. As soon as MN attaches to the new AP, link state change may trigger DAD operation for the link-local and global addresses. A DAD operation for a link-local address is required since address collision is possible between MN, MAG and all MNs attached to the same MAG. The DAD operation is a very time consuming procedure and may significantly increase handover time. Typically, the DAD operation time is around one second and sometimes can be much more. However, PMIPv6 introduces a per-MN prefix model in which every MN is assigned a unique home network prefix (HNP) [4]. This approach may guarantee address uniqueness. Thus, the DAD processing for global address may not be necessary. In PMIPv6 it is required that MAG should advertise MN's HNP in RA messages sent on its serving link. However, these advertisements can be solicited and unsolicited as well. According to the neighbor discovery protocol [10], MN should wait a random time between 0 and 1 second before sending RS message. On reception

RS message, MAG has to delay RA by a random time between 0 and 500ms. Thus, for solicited router advertisements the overall delay is the sum of RS delay, RA delay and round-trip time between MN and MAG. In many cases it is preferred to configure MAG to send unsolicited router advertisements. Comparing with MIPv6, in PMIPv6 there is no strict requirement to send RAs frequently since PMIPv6 doesn't perform movement detection. However, MAG still has to backoff sending RAs.

3.2 Low Packet Loss and Latency Proactive Handover Scheme for PMIPv6 with MIH

In this paper, we propose a low packet loss and minimized latency proactive handover scheme for PMIPv6 by using MIH services. Based on the handover analysis made above we underlined the major problems that may cause high handover delay in PMIPv6. As it was already mentioned one of the most time consuming procedure during L2 handover is channel scanning time. By using MIH service we reduced the channel scanning time performed by MN [7]. Fig. 1 depicts the sequence diagram of the proposed low latency proactive handover scheme for PMIPv6 with MIH. At the very beginning, MIHF of the MN registers to MAG. The registration is done based on the MIHF ID and link identifiers. After registration MAG may receive event notification from MN such as L2 event triggers. Furthermore, MAG retrieves information about neighbor access networks to which attached MNs may handover. This retrieval is done through the *MIH_Get_Information* request-response exchange with Media Independent Information Server (MIIS). When the link conditions between MN and serving MAG deteriorate and the signal becomes weak, the MIES of MAG receives *MIH_Link_Going_Down* event notification message. MAG searches its cache to retrieve information about neighbor access networks including channels on which MN has to perform scanning. This information is delivered to MN through the *MIH_Link_Action* request.

Once the scanning is finished by MN, MAG starts to collect information required for target point of service (PoS) selection. MAG performs request to the candidate PoSes for availability of resource and bandwidth to provide QoS through the *MIH_N2N_HO_Query_Resource* request-response exchange. Based on the information received from a candidate networks MAG makes handover decision and selects a new MAG (nMAG). When receiving resource request, candidate MAG can check the MN's address uniqueness, since MAG has information related to every MN currently attached to it.

After handover decision is made, pMAG requests the handover to the nMAG by sending *MIH_N2N_HO_Commit* request. nMAG replies to the request message and retrieves MN's profile information (MN_ID, MN_IF_ID, ATT, etc.). When pMAG get response message from nMAG indicating successful operation, it sends *MIH_Net_HO_Commit* request message which includes the target PoS. At this point, MN may attach to the selected PoS. At the same time, pMAG sends Fast PBU (FPBU) message to LMA de-registering the MN and triggering pre-registration with nMAG. LMA performs pre-registration by sending Handover Initiate (HI) to nMAG which includes pMAG address. nMAG replies with Handover Acknowledge (HACK) indicating successful pre-registration. Since pre-registration is successful, nMAG may

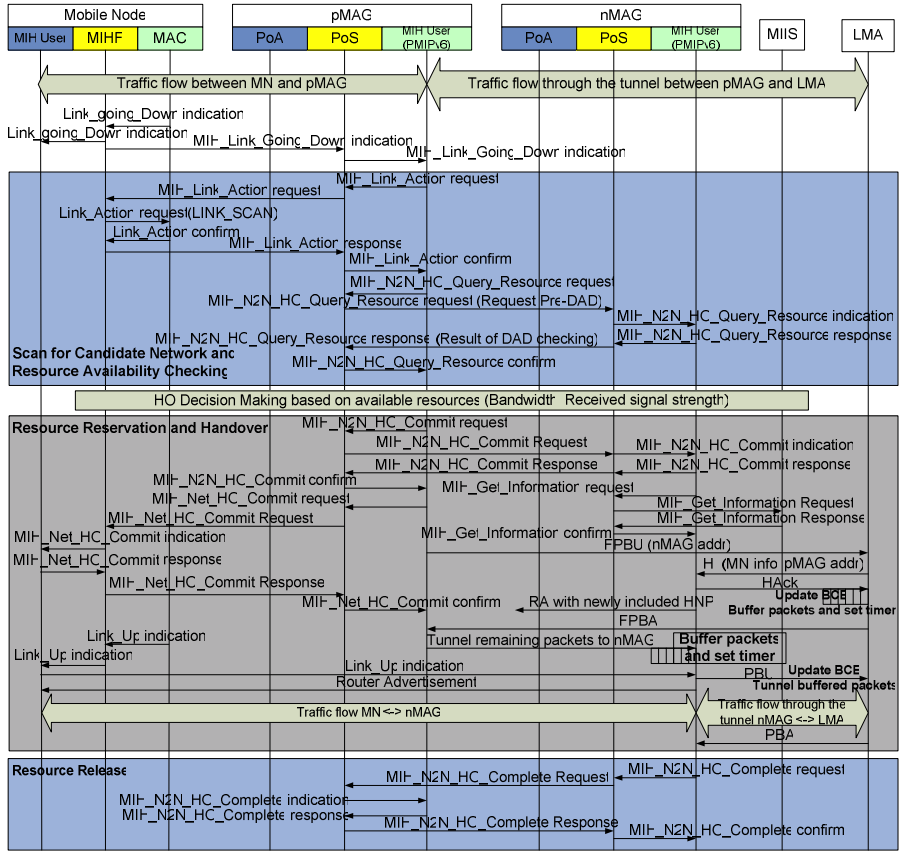


Fig. 1. Sequence diagram of low latency proactive handover for PMIPv6 with MIH

```

MAG mobility message process ()
{
CASE Received message OF
PBA      : Perform default processing for PMIPv6
HI        : Check the availability of PMIPv6 service
IF the request is accepted THEN
Set the HNP to Router Advertisement message
Add entry to binding update table
ENDIF
Send HACK to LMA
FPACK : Tunnel remained packets to nMAG
New MN attachment :
IF the MN is pre-registered THEN
Send PBU update message to LMA
Transmit the buffered data
ENDIF
ENDCASE
}

LMA mobility message process ()
{
CASE Received message OF
PBU      : Perform default processing for PMIPv6
Remove the timer for pre-registered BCE
Update lifetime
FPBU     : IF lifetime 0 THEN
Send HI to target MAG
ENDIF
HACK : IF the request is accepted THEN
Update BCE ()
Set the timer of Pre-registered BCE
Start buffering and set timer
ENDIF
Send FPBA to MAG
}

```

Fig. 2. Pseudo code of mobility message processing

now include MN's HNP to the router advertisements sent on its link. Furthermore, a bi-directional tunnel can be established between pMAG and nMAG since both of them have address information of each other. When receiving HACK message LMA

updates binding cache table and starts buffering packets destined to the MN. Additionally, it sets timer for the buffered data since in worst case handover may not be completed. LMA replies with fast PBA (FPBA) to the pMAG indicating successful de-registration. At this point, pMAG may tunnel remaining packets to the nMAG minimizing the packet loss that may occur due to the MN's unreachability. nMAG buffers delivered packets and sets the timer for the buffered data. During MN's attachment to the nMAG, *Link_UP* indication will be delivered triggering PBU transmission to LMA. After reception of PBU, LMA updates the lifetime for the MN's entry in binding cache table and starts transmission of the buffered data through the tunnel between LMA and MAG. In our proposed scheme, we consider that an unsolicited router advertisement method. When the first RA message is transmitted after MN's attachment, MAG may start to send data to the MN. Fig. 2 shows the pseudo code of mobility message processing.

3.3 Analysis of Handover Latency in the Proposed Scheme

The overall handover latency consists of the L2 and L3 operations. In our proposed scheme, we reduce the time taken for scanning by informing MN about the channel information of neighbor APs. This allows the MN to perform scanning only on the certain number of channels which can significantly reduce the scanning time. Thus, the L2 handover delay can be expressed as:

$$T'_{L2} = T'_{scan} + T_{auth} + T_r \quad (3)$$

However, the scanning time also depends on the type of scanning is used. There are two types of scanning defined: active and passive. In active mode, station sends probe requests and receives probe response if any AP is available on certain channel. In passive mode, a station "listens" the channel for possible beacon frames generated by AP. The handover time in active mode is usually less than in passive mode. The passive mode of operation depends strictly on the period of beacon generation interval. However, this can provide better battery saving than active mode of operation.

Equation (2) shows the major parts of L3 handover delay that should be taken into consideration in PMIPv6. Since our proposed scheme performs in proactive mode, the number of operations should not be considered for overall handover latency. The time needed to obtain MN's profile information can be omitted since nMAG performs this information retrieval prior to MN's actual attachment. The time required to perform DAD can also be omitted, since nMAG performs a pre-DAD procedure during available resource negotiation with pMAG. The RS message backoff and transmission time may not be considered because we only consider unsolicited approach. The signaling message exchange (FPBU <-> FPBA, HI <-> HAcK) time during pre-registration should not be considered, since this negotiation is made before MN's attachment. The last PBA message sent from the LMA may not be considered, since MN was already pre-registered and there is no need to confirm it at the nMAG.

There are only two things to consider in the proposed scheme for MN's handover at L3. First of all, when MN attaches to the nMAG and delivers *Link_UP* event notification, nMAG has to send PBU message to the LMA updating the lifetime of the entry in the binding cache table of the LMA and triggering transmission of buffered data for the MN. Another thing to consider is RA message generation delay. Thus, the overall handover delay at L3 can be expressed as:

$$T'_{L3} = T_{PBU} + T_{RA} \quad (4)$$

As equations (3) and (4) show, the overall handover latency can be reduced significantly in the proposed low latency proactive PMIPv6 handover with MIH scheme.

4 Performance Evaluation and Analysis

We analyze the performance of our proposed scheme and compare it with PMIPv6 through the simulations performed in network simulator. We implemented the proposed low latency proactive PMIPv6 scheme with MIH in OPNET network simulator [12]. Although the overall integration of our modules is in the debugging stage, we present performance analysis and discussions available now. We provide analysis of delay in PMIPv6 and our proactive approach. Fig. 3 shows the simulation scenario for the evaluation.

Simulation scenario shows that there is a single LMA and two MAGs within one PMIPv6 domain. MN performs horizontal handover from pMAG to nMAG and while moving from BSS1 to BSS2. Correspondent node (CN) is located outside of the PMIPv6 domain somewhere in the Internet. IEEE 802.11b wireless LAN was used as the access network technology. VoIP application was simulated for communication between MN and CN with a constant bit rate of 64 kbps (PCM).

Table 1 shows the simulation result for the proposed scheme and PMIPv6. The three parameters are related to the L2 handover time. The authentication is simply round-trip time between MN and AP, since an open system authentication was used. For the re-association we also considered only round-trip time and omitted IAPP message exchange because it is made through the wired network with very low delay.

The scanning time was significantly reduced because the MN doesn't have to perform scanning for every channel. The time taken to get MN's profile is simply information retrieval and doesn't add much time to the overall handover. The time for PBU and PBA transmission is the propagation delay between MAG and LMA.

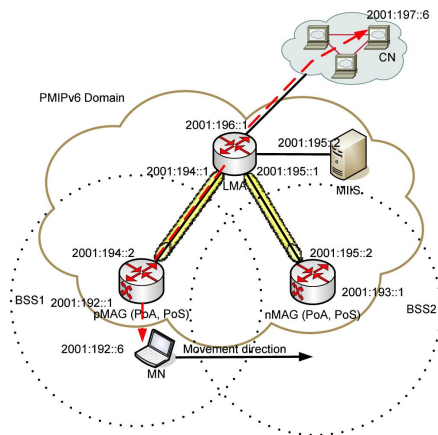


Fig. 3. Simulation scenario

Table 1. Performance evaluation results

Parameter	PMIPv6	Proposed Proactive PMIPv6 with MIH
T_{auth}	0.6ms	0.6ms
T_r	0.6ms	0.6ms
T_{scan}	250ms	25ms
T_{MN}	0.1ms	
T_{PBU}	11ms	11ms
T_{PBA}	11ms	
T_{DAD}	1.07s	
R_{RA}	30ms	30ms
<i>Total</i>	1373.3ms	67.2ms

This delay is proportional to the distance between LMA and MAG and communication link speed. The highest delay is experienced by MN when performing DAD. Since in our proposal we provide a pre-DAD scheme, the time needed for DAD can be omitted. However, in PMIPv6 a DAD scheme may not be used since PMIPv6 introduces per-MN prefix allocation. Additionally, optimistic DAD scheme can be applied. Even without DAD consideration the proposed scheme can provide significant improvement in handover latency. As it was already mentioned, we applied unsolicited RA approach. However, we cannot send RA message immediately after MN's attachment. Additional backoff time is necessary in order to give time to MN to configure its interface. Based on our investigations and [11] we suggest setting this time in the interval 30-50 ms.

Performance evaluation results show that the overall handover latency can be significantly reduced by utilizing MIH services and performing handover in proactive manner.

5 Conclusion

In this paper, we proposed a novel low latency proactive handover procedure for PMIPv6 using MIH services. By using MIH services we provide a solution to reduce the overall channel scanning time which can substantially reduce the L2 handover delay. By using MIH services, a QoS-aware target PoS selection may be performed based on the information obtained from the negotiation. A pre-DAD scheme was proposed, which is performed by pMAG prior to MN's attachment. By utilizing ideas of FMIPv6 [2] the proactive handover scheme for PMIPv6 was proposed. An efficient buffering scheme allows reducing a packet loss during handover. A method to deliver packets from pMAG to nMAG through the tunnel was explained. By using this approach the packets that could be possibly lost at pMAG are delivered to nMAG.

The future work includes detailed analysis of PMIPv6 performance for multihoming. The vertical handover for PMIPv6 with MIH will be studied deeply investigating the major problems that occur during this procedure. Our major goal is to provide a seamless service with QoS guarantees in next generation mobile networks.

References

1. Johnson, D., Perkins, C., Arkko, J.: Mobility Support in IPv6. RFC 3775 (June 2004)
2. Koodly, R.: Fast Handovers for Mobile IPv6. RFC 4068 (July 2005)
3. Soliman, H., Flarion, Castelluccia, C., El Malki, K., Bellier, L.: Hierarchical Mobile IPv6 Mobility Management, RFC 4140 (August 2005)
4. Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., Patil, B.: Proxy Mobile Ipv6. draft-ietf-netlmm-proxymip6-17.txt, May 27 (2008)
5. IEEE P802.21/D10.0 Draft standard for Local and Metropolitan Area Networks: Media Independent Handover Services (April 2008)
6. Kempf, J.: Problem Statement for Network-Based Localized Mobility Management (NetLMM). RFC 4830 (April 2007)
7. Kim, B.K., Jung, Y.C., Kim, I., Kim, Y.T.: Enhanced FMIPv4 Horizontal Handover with Minimized Channel Scanning Time based on Media Independent Handover (MIH). In: BcN 2008 (April 2008)
8. Chou, C.-T., Shin, K.G.: An Enhanced Inter-Access Point Protocol for Uniform Intra and Intersubnet Handoffs. IEEE Transactions on Mobile Computing 4(4) (July/August 2005)
9. Wu, H., Tan, K., Zhang, Y., Zhang, Q.: Proactive Scan: Fast Handoff with Smart Triggers for 802.11 Wireless LAN. In: INFOCOM (2007)
10. Narten, T., Nordmark, E., Simpson, W., Soliman, H.: Neighbor Discovery for IP version 6 (IPv6), RFC 4861 (September 2007)
11. Mueller, C.M., Blume, O.: Network-based Mobility with Proxy Mobile IPv6. In: IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (2007)
12. <http://www.opnet.com>

Simple Modeling for QoS Management of IEEE 802.11 DCF

Takashi Satake

NTT Service Integration Laboratories
Nippon Telegraph and Telephone Corporation
9-11, Midori-Cho 3-Chome, Musashino-Shi, Tokyo 180-8585, Japan
`satake.takashi@lab.ntt.co.jp`

Abstract. As widely seen in various private networks and public wireless hotspots, the spread of wireless LAN based on IEEE 802.11 has reached a peak, and its IEEE standard continues to be improved toward 802.11n. On the other hand, the quantitative method of evaluating wireless LAN has not caught up with the evolution of its technology and is still under development. Indeed, some design or management know-how, such as a placement design of access points (APs) aimed to avoid radio interference or traffic congestion, has been accumulated, while some analytical approaches based on a strict mathematical model are studied. However, we cannot say that they are practical and sufficient for actual network management situation. In this paper, a simple and practical performance evaluation model of IEEE 802.11 DCF based on the $M/M/1/K$ queue, to contribute to establishing efficient QoS management of wireless LAN networks is presented. The relationship between the packet loss ratio, which is one of the important metrics in QoS management, and the MAC retry ratio, whose measurement is comparatively easier, is demonstrated in this model. Some numerical examples are described, and their effectiveness is discussed. In addition, some experimental evaluations corroborating the numerical analysis, are presented.

Keywords: IEEE 802.11 DCF, $M/M/1/K$ queue.

1 Introduction

The IEEE 802.11 wireless local area network (LAN) [1] has spread rapidly and has become popular everywhere in the office or home, and in public spaces like stations, airports, and restaurants. Its IEEE standard continues to be improved toward 802.11n [3].

On the other hand, the quantitative evaluation method has not caught up with the evolution of the technology and is under development. Indeed, some design or management know-how, such as placement design of APs aimed to avoid radio interference or traffic congestion, has been accumulated [4],[5], while some analytical approaches based on a strict mathematical model are studied (see section 2.3). However, we cannot say that they are practical and sufficient for actual network management situations.

Many real-time applications like VoIP or streaming communication services are provided. As broadband access services like xDSL and FTTH have become available, these applications provide real-time communication tools. We expect that real-time applications over wireless LAN will also be used more often in home networks and enterprise networks.

The IEEE 802.11g wireless LAN system [2] is the most popular, and the 2.4-GHz band is allocated to it. This band is reserved for ISM (Industrial Scientific Medical) fields, so wireless LAN systems may be subject to radio interference from other devices. Frequency bands allocated for wireless LAN are limited, so the channel for each wireless access point needs to be allocated efficiently.

However, some performance problems are caused by CSMA/CA channel access protocols as wireless service environments become diversified. For example, there is a hidden terminal problem (Fig 1). In this situation, sharing resources fairly is not easy because many stations are operating on the same channel. Real-time, interactive applications like VoIP, IPTV, or streaming services are sensitive to network performance variation, so higher quality has been required in such wireless LAN environments. However, current products are not intended to deal with various service environments and are not sufficiently functional to meet the level of quality needed by users.

In this paper, a simple and practical performance evaluation model of IEEE 802.11 DCF based on the $M/M/1/K$ queue is presented, to contribute to establishing efficient QoS management of wireless LAN networks. The relationship between the packet loss ratio, which is one of the important metrics in QoS management, and the MAC retry ratio, whose measurement is comparatively easier, is demonstrated in this model. We can expect to detect quality degradation of applications by using the metric. Some numerical examples are described and their effectiveness is discussed. In addition, some experimental evaluations in an actual wireless environment, which corroborate the effectiveness of the proposed method, are presented.

The remainder of this paper is organized as follows. In Section 2, an overview of the CSMA/CA mechanism is presented, and problems in the mechanism are illustrated. The details of the proposed model are described in Section 3. The experiments and their results are presented in Section 4. Finally, conclusions are given in Section 5.

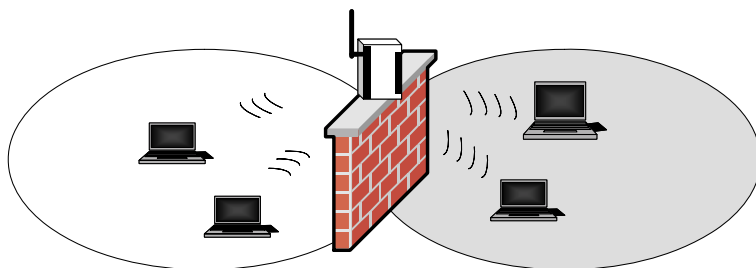


Fig. 1. Typical hidden-terminal problem

2 QoS Management of Wireless LAN

2.1 CSMA/CA Mechanism

The basic access mechanism of the IEEE 802.11 MAC is the Distributed Coordination Function (DCF) [6], which is known as Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). In the CSMA/CA mechanism (Fig. 2), a station starting to transfer packets monitors the medium before the transmission. If the channel is busy, the station defers its transmission to a later time. If the channel is unoccupied for a specified time, namely the Distribution Inter Frame Space (DIFS), the station generates a random backoff period for an additional deferral time before initiating transmission. This process minimizes the number of collisions during contention between multiple stations. The receiving station sends an Acknowledgement (ACK) packet after a specified time called the Short Inter Frame Space (SIFS). If the sender does not receive the ACK, the MAC layer retransmits the frame until it receives the ACK or discards the frame after the number of retransmissions reaches its limit.

2.2 QoS Degradation over Wireless LAN

In some environments, when multiple stations establish communications, not all of the stations will be in range to receive packets, and packets from the stations, which cannot sense each other, may collide. This is the above-mentioned hidden-terminal problem [7].

In considering the mechanism of quality degradation when the CSMA/CA mechanism does not work well, two points are usually the focus from the viewpoint of QoS management. One point is transmission delay and the other point is packet discard. In actual environments, many packets may not be transmitted at the right time because signal strength becomes weak in the propagation, and there is a failure to sense carriers. In those cases, frequent collisions cause

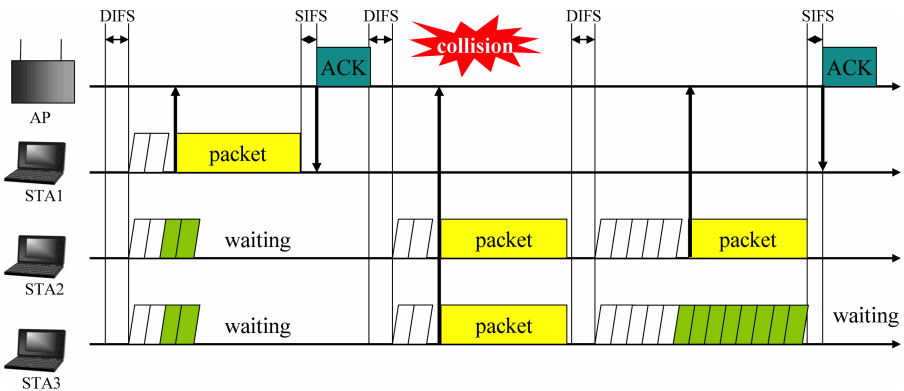


Fig. 2. CSMA/CA mechanism

serious degradation of network performance, and both points are closely related. Therefore, defining the parameter for CSMA/CA effectiveness is necessary when we evaluate performance on wireless LANs.

2.3 Related Work on Performance Modeling

In many studies on the CSMA/CA mechanism [8],[9], transmission delays of packets are evaluated under increasing traffic conditions. Theoretical calculation and simulation results indicate some approximated delays in this mechanism. These approaches suppose the channel-access mechanism in ideal environments, so the probability of packet collisions is theoretically small.

However, the effectiveness of the protocol is not always assured under actual wireless environments, and appropriate metrics should be considered for some wireless environments. Exponential backoff is modeled, and throughput is analyzed, but that does not consider the existence of a MAC buffer [6]. Another model is based on the $M/G/1/K$ queue considering the existence of a MAC buffer [10]. This is a strict mathematical modeling, but the iterative process in its calculation is complex, and unfortunately, its convergence is not guaranteed.

2.4 Metrics for Detection of QoS Degradation: p_{re}

A procedure for detecting quality degradation is necessary to elucidate the mechanism in actual environments. In wireless LAN systems, judging whether collisions happen or not is not easy, so some metrics, which can be easily measured, should be defined to monitor signs of degradation. We propose to use the information in the 802.11 MAC frame field.

Frame error, data-rate mode, and retries, which show the radio characteristics, are described in each packet. If transmitted packets have an error or the data-rate mode changes to a slow rate, these packets are retransmitted. The number of retry packets during a given time interval is counted, and statistics are used as criteria of judgment in reacting.

We thus characterize the quality degradation by the MAC retry ratio [11], which is defined by the number of retries for packets transmitted for a measuring interval. Let n_T be the number of packets transmitted for the measuring interval T , and $n_{re,T}$ be the number of retransmitted packets. The MAC retry ratio is then defined by

$$p_{re} = \frac{n_{re,T}}{n_T}.$$

Here, we can simply count the number of retransmissions during interval T .

3 Approximation of CSMA/CA Performance Based on $M/M/1/K$ Queue

In this section, a performance evaluation model of IEEE 802.11 DCF based on the $M/M/1/K$ queue is introduced using basic queueing theory [12], and the

relationship between the packet loss ratio and the MAC retry ratio, p_{re} , which has been introduced in 2.4, is demonstrated.

3.1 Derivations

Average Waiting Time in i-th Backoff Stage: t_i At first, the time to stay in each backoff stage is introduced. The occupancy ratio of background traffic is represented by σ . In other words, σ is the occupancy ratio, which determines how much a packet is kept waiting under transmission of another packet,

$$\begin{aligned} t_i &= \sum_{j=0}^{\infty} jA\sigma^j(1-\sigma) + B_i \\ &= \frac{\sigma}{1-\sigma}A + B_i. \end{aligned} \quad (1)$$

Here, A is a fixed time taken to wait for another packet to be transmitted, and B_i is a time taken to complete transmitting the packet in the i -th stage ¹.

Average Waiting Time: h Here, p_{re} is the probability that the transmitted packet collides with another packet. If the transmission has failed, then the packet goes to the next backoff stage until the number of retries exceeds the limit k . Otherwise, the packet leaves the buffer. Thus, the total waiting time for the queueing system is

$$\begin{aligned} h &= \sum_{l=1}^k p_{re}^{l-1}(1-p_{re}) \cdot t_l + p_{re}^k \cdot t_k \\ &= \sum_{l=1}^k p_{re}^{l-1}(1-p_{re}) \cdot \left(\frac{\sigma}{1-\sigma}A + B_l\right) + p_{re}^k \cdot \left(\frac{\sigma}{1-\sigma}A + B_k\right). \end{aligned} \quad (2)$$

Here, though p_{re} actually depends on σ , we suppose p_{re} and σ are independent to simplify the analytical process.

Offered Traffic: ρ Let the Poisson arrival rate of the packet be λ ,

$$\rho = \lambda h. \quad (3)$$

Here, the dimension of ρ is the Erlang unit.

¹ For example, by the method reported in [8],[9],

$$\begin{aligned} A &= T_c + T_{bg} + T_a + 2SIFS + DIFS \\ B_i &= 2^{i-1}CWmin \cdot SlotTime + DIFS + T_d. \end{aligned}$$

Here, $CWmin$ means the minimum contention window size. CTS transfer time is T_c , data packet transfer time is T_d , background packet transfer time is T_{bg} , and Ack transfer time is T_a .

Buffer Overflow Ratio: b_{ovf} The buffer overflow ratio of the $M/M/1/K$ queue is as follows when the buffer length is m [12].

$$b_{ovf} = \frac{1 - \rho}{1 - \rho^m} \rho^{m-1} \quad (4)$$

Packet Loss Ratio: b Except for losses caused by the overflow from the MAC buffer, losses caused by collisions or interferences after the packet has left the buffer can occur. This is considered, and finally, the packet loss ratio is given by

$$b = b_{ovf} + p_{re}^k (1 - b_{ovf}). \quad (5)$$

3.2 Numerical Examples

Some calculation results of the above-mentioned analytical formula (1)-(5) obtained under the values of Table 1 are shown in Figure 3. The system values in Table 1 were determined from 802.11 standards. The packet loss ratio and its differential coefficient increase if retransmission increases under the condition that σ is constant, as shown in Figure 3.

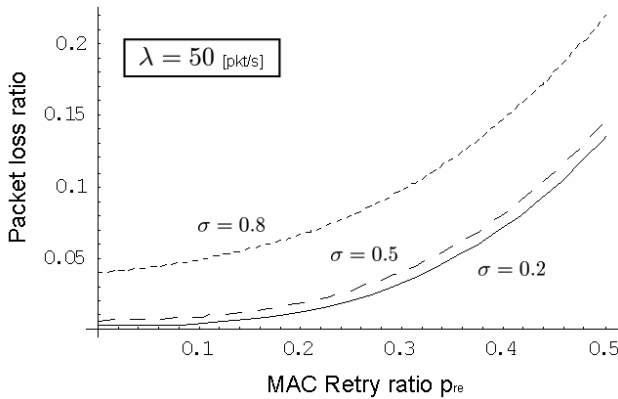


Fig. 3. Relationship between packet loss ratio and MAC retry ratio

Table 1. Calculation parameters

Parameter	Value
Slot Time	$20 \mu s$
SIFS	$10 \mu s$
DIFS	$50 \mu s$
CWmin	31
Retry limit k	3
Buffer length m	2

4 Experiments in Actual Environment

As a first step in the verification of the proposed model, experiments in actual wireless LAN environments were performed.

4.1 Experimental Conditions

The experimental system was composed of two different systems operating on the same channel (2.447 GHz), as shown in Figure 4. In system 1, data packets were transmitted to wireless ST2 at a constant interval from Ethernet ST1. An IP packet has three lengths: 1480, 1000, and 520 bytes, and there were three transmission intervals: 10, 20, and 30 ms, respectively, so total byte throughputs were almost identical. In system 2, wireless ST3 and ST4 transmitted data packets as background traffic. The length of an IP packet in system 2 was 1500 bytes under the UDP.

Packets were transmitted and received between ST1 and ST2. The IP Packet loss ratio is defined by the ratio of the total number of lost packet outcomes divided by the total number of transmitted IP packets. IP packet loss ratio was estimated by checking the RTP IDs of transmitted and received packets.

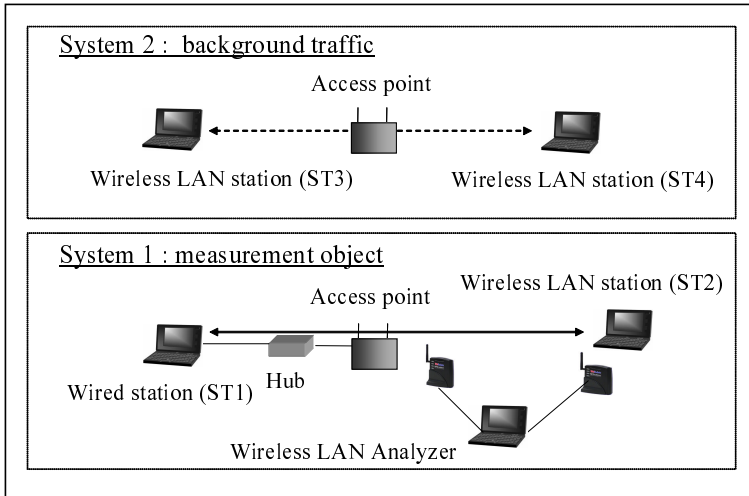


Fig. 4. Wireless LAN experimental systems

Signal strengths of System 1 and System 2 were monitored by a wireless LAN analyzer [13]. As a criterion to obtain less variable results, we used the ratio of desired signal (system 1) divided by undesired signal (system 2) and set that to 15 dB. At the same time, the number of retransmitted packets was counted. This experiment was performed for one minute and averaged over ten values.

4.2 Results and Discussion

In Figure 5, a graph of data from the experiment and the analytical curves based on the proposed model is shown. We can confirm that each point plotted by measured data agrees well with each theoretical curve calculated by this model. Here, the condition in which retry limit $k = 3$ and buffer length $m = 2$ was the most suitable to fit the measured data to the analysis. This is almost identical to normal implementations of wireless LAN cards and personal computers. A correlation between σ and p_{re} is considered in this analysis because background traffic generation uses the same 802.11 protocol.

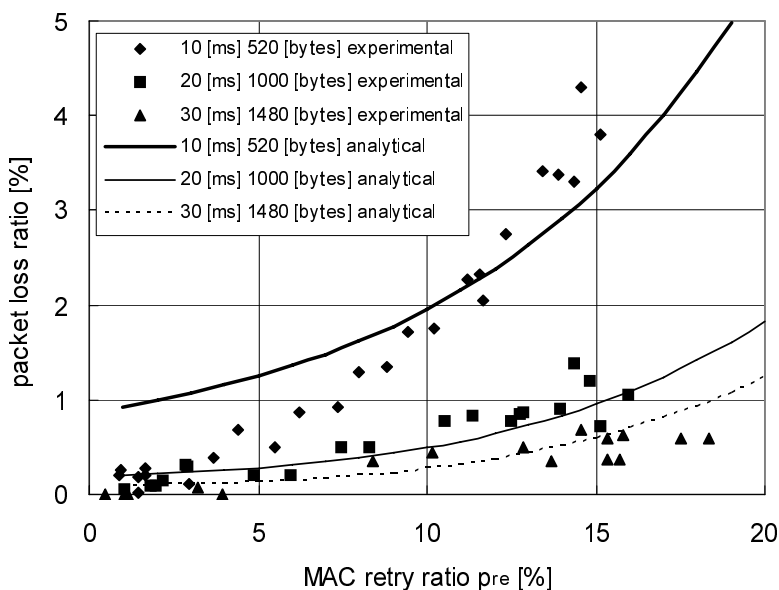


Fig. 5. Comparison between analysis and experiment

5 Conclusion

In our attempt to guarantee QoS for wireless LAN networks, we paid attention to the MAC retry ratio as a metric for detecting quality degradation. The mechanism of quality degradation in 802.11 environments was elucidated, and the packet loss ratio was evaluated using a simple model of a CSMA/CA mechanism. Then, the relationship between the packet loss ratio and the MAC retry ratio was demonstrated. The experiment corroborates the validity of the proposed numerical model. In the standard for QoS management, especially in ITU-T Y.1540 [14] and its series, the packet loss ratio is defined as IP quality metrics, and particular values for each service application are also recommended. These results suggest that the MAC retry ratio is applicable to detection of QoS degradation, and could be used as a criterion for QoS management in wireless LAN networks.

In the near future, these results are expected to facilitate rapid and easy detection of QoS degradation in a wireless LAN access network and MAC layer enhancement for a robust QoS management scheme under various service environments.

Acknowledgements

The author would like to thank Mr. Tatsuya Matsukawa and Mr. Jiro Yamada for their contributions to the early part of the study, and Mr. Isami Nakajima, Dr. Haruhisa Hasegawa, and Mr. Hideaki Yoshino for their many helpful suggestions.

References

1. IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, ISO/IEC 8802-11: 1999(E) (August 1999)
2. IEEE Standard 802.11g, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 4: Further High Data Rate Extension in the 2.4GHz Band (June 2003)
3. IEEE Standard 802.11n, Draft 2.0, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification: Enhancements for Higher Throughput (February 2007)
4. Eisenblatter, A., Geerdes, H., Siomina, I.: Integrated Access Point Placement and Channel Assignment for Wireless LANs in an Indoor Office Environment. In: IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, WoWMoM 2007, pp. 1–10 (June 2007)
5. Vanhatupa, T., Hannikainen, M., Hamalainen, T.D.: Genetic Algorithm to Optimize Node Placement and Configuration for WLAN Planning. In: 4th International Symposium on Wireless Communication Systems, ISWCS 2007, pp. 612–616 (October 2007)
6. Bianchi, G.: Performance Analysis of the IEEE802.11 Distributed Coordination Function. *IEEE Journal of Selected Areas in Telecommunications, Wireless series* 18(3) (March 2000)
7. Heusse, M., Rousseau, F., Berger-Sabbatel, G., Duda, A.: Performance Anomaly of 802.11b. In: IEEE INFOCOM 2003 (2003)
8. Xylomenos, G., Polyzos, G.C.: TCP and UDP Performance over a Wireless LAN. *IEEE Wireless Communications* (June 2002)
9. Bianchi, G., Tinnirello, I.: Kalman Filter Estimation of the Number of Competing Terminals in an IEEE802.11 network. In: Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies on INFOCOM 2003 (2003)
10. Zheng, Y., Lu, K., Wu, D., Fang, Y.: Performance Analysis of IEEE 802.11 DCF in Imperfect Channels. *IEEE Trans. Vehicular Technology* 55(5), 535–547 (2006)
11. Matsukawa, T., Satake, T., Yamada, J.: Metrics for Detection of QoS Degradation over Wireless LAN. In: International Symposium on Applications and the Internet SAINT 2005 (February 2005)
12. Gross, D., Harris, C.M.: *Fundamentals of Queueing Theory*, 3rd edn. Wiley, New York (1998)
13. <http://www.wildpackets.com/>
14. ITU-T Recommendation ITU-T Y.1540, Internet Protocol Data Communication Service-IP Packet Transfer and Availability Performance Parameters (February 1999)

Take the Challenge of IP Evolution and OAM Solution

Lihong Wei and Lingshan Kong

Network Department of China Mobile

weilihong@chinamobile.com, konglingshan@chinamobile.com

Abstract. During the process of the strategic transformation of the telecom industry, the Mobile IP Network is telecom operators' only choice as they strive to provide diversified services and try to survive and grow. The Mobile IP Network, with its characteristics such as low cost and easy to deploy with new services, is now promoting an IP revolution step by step. To realize a smooth E2E transformation to the Mobile IP Network, operators must facing some of the IP evolution challenges for OAM. To overcome those challenges, what China Mobile does on network management, service and customer experience support is discussed in the paper.

Keywords: IP Network, soft switch, network management, Service support, customer experience support.

1 The Driving Forces to Mobile IP Networks

China Mobile started carrying out a comprehensive strategic transformation to IP networks in 2004, when IP-based softswitch tandem was first constructed¹. Following that, the IP end office voices and the IP-based BTSs were put into trial applications in 2006 and 2007 respectively. Internationally, companies such as NTT, BT, T-Mobile, and ORANGE also stepped on their way to the mobile IP network one after another. The mobile IP network has been gradually recognized as a mode that can meet the network and service convergence requirements during the industry transformation. The following driving forces are pushing the mobile IP network in place.

The major driving force is the surviving pressure of telecom operators. To survive the increasingly severe telecom competition, operators are driven to provide more diversified services at lower cost, which can only be realized through extendable networks with powerful vitality in the long term. Currently, however, with the growing of the mobile services, the returns of the TDM investment are actually decreasing. Operators are thus forced to turn to the mobile IP network, which is outstanding in both service bearing capability and extendibility.

Secondly, the demand of fast response to new service requirements is an important factor. The telecom industry is now faced with various challenges, varying from new service competition, low-cost Internet voice services, to increasing service volumes with decreasing economic benefits. Moreover, traditional IP networks can provide required new services very quickly through the mere update of softwares, which constitutes a tremendous impact on the market and thus forces the operators to renew their concepts towards the only choice available: mobile IP networks.

Thirdly, the emerging equipment providers strive for further development, which serves as the original driving force to the mobile IP network. As the important node in the industry ecological chain, equipment providers can only grow and benefit by providing IP equipments in favor of service convergence and network convergence, and with broad bandwidth and wide adaptability. It is this need of the ecological chain to grow that makes it inevitable for the telecom industry to realize the strategic IP transformation.

2 The Gradual Process of Mobile IP Networks²

The process of developing mobile IP networks covers three aspects with respective aims. Terminal IP transformation is to support, and focus on, the individual experiences in multiple terminal services through software installation. Service IP transformation aims to realize service convergence, overcoming the shortcomings of the original network services, such as long service deployment cycle and poor adaptability. Finally, network IP transformation tries to meet various bandwidth and quality requirements at a lower cost, and realize separate bearing and controlling functions, which is a breakthrough in bandwidth restriction. All these make it possible for network IP to provide higher QoS and reliability.

The network IP transformation is a step-by-step process starting from core networks to border equipments. Specifically, the first step is to realize the IP bearer in long distance networks, by importing the R4 core network structure with separate bearing and controlling functions. The second step is to accomplish the softswitch transformation in end offices and gateway exchanges, and lastly to achieve the E2E All-IP transformation including transmission networks and wireless access networks.

During the process of developing IP networks, Chinese companies progress even faster. Operators such as China Mobile and China Telecom have already completed the construction of IP bearer networks and started large scale softswitch applications. As estimated, traditional core networks with, for example, GSM switch equipments, will finish the IP transformation within a mere couple of years, but for the mobile network, it takes a considerably long period to accomplish the E2E IP transformation.

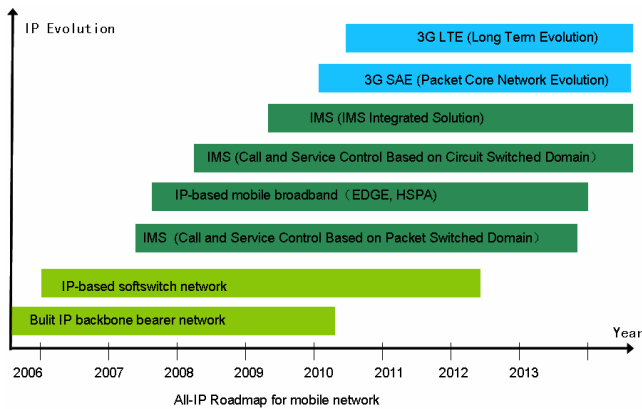


Fig. 1. All-IP evolution roadmap of mobile network

3 The Technology Trends in Mobile IP Networks

The technology trends in developing mobile IP networks are observed as follows.

The IP technology will penetrate into every node in the network. With the extension of IP technology application, the seamless convergence of networks and services in fixed and mobile networks can be realized by the IP transformation of the core network softswitch, integrated access network, IMS platform, services and terminals, which can be combined through a flat service access structure. No matter for traditional switch networks, wireless networks, access networks, data networks or other specific features, the IP-based open platform and standards can gradually enable the comprehensive bearer of them through the mass saturation of the IP technology.

The reliability technology is gradually trending towards a sound and stable state through integrated planning. Specific protocols and function elements need to be designed separately for different network layers to solve the issues in traditional IP networks. Examples are: out-band console interface control capability improves the control reliability in the IP/IT system; fast convergence technologies like FRR improve the speed of network recovery; the graceful restart capability improves the network stability; dual router engines improve the equipment reliability; and, MSC POOL improves the service security of the softswitch layer. All these new features are under gradual improvement and achieving the stability through practice.

The schemes of IP network security and QoS will be more practical. In IP networks which are based on open structures and best effort forwarding systems, security and quality assurance have always been the weakest links. With the deepening of the IP transformation, the industry has come to realize that IP security is an integrated system requiring all-directional attention to each layer from every angle. Security requirements can be met by adopting different security strategies, as well as security domain allocation, for defense and isolation, and by taking all-round management, technology and design measures. For SLA quality assurance, it's preferred to adopt light load and redundancy integration systems in the core network layer, deploy QoS strategy forwarding in the convergence layer, and deploy colorizing and speed limiting in the access layer. From the application angle, concepts such as service perception and lower layer probing will be promoted to realize mutual info exchange. With the popularization of all forms of IP applications in the industry, the schemes for security and quality assurance will be more and more practical and sound.

4 IP Evolution Challenges on Networks' Management

To realize a smooth E2E transformation to the Mobile IP Network, operators must facing the IP evolution challenges for management.

Firstly, Locating fault is difficult since service is isolated from network. In Legacy TDM network fault locating is easy for TDM resource is static and fault locating is easy. While in IP network, the carrier of service depends on dynamic route. It's hard to locate fault in NE when problem appeared in media flow. So IP network fault locating is difficult. No resource reservation for non-connection oriented IP networks is the reason that QoS assurance is hard in ip network. To solve this problem, we need integrated Resource Management System to isolate alarm and locate fault and build connection between Service and Network Tier to assure QoS from the service point of view.

Secondly, Rapid increase in service types creates Service Management challenges. As of 2007, more than 100 kinds of data services have been provided by China Mobile. It is difficult problem for operator to manage the complex network and service profiles. In new networks, services available to customers are not implemented by a single network or technology. There are complicated, trans-disciplinary, comprehensive relations between services and networks which might be a combination or one bearing on another. Service management is aimed at implementing end-to-end management.

The last, operator meets the challenge to offer a quality customer experience. Customer complains about bad voice quality or MMS, but at the same time network KPI result is OK. VIP customer complains about voice interruption with noises while Call Quality test is OK, etc. We have an example here for one customer frequent complaints MMS services, such as terminals do not support MMS, wrong default configuration on terminals, complicated configuration, Customers' habitual behavior (e.g., not receiving MMS messages).

5 Take Challenge of IP Networks' OAM

Comprehensive end-to-end OAM solution is required for IP networks. Key factors need to be considered when we facing this challenge.

Adapt service to consumer behavior. Bundling services and subscriber self-provisioning are key initiatives for success. These requirements are highly dependent on accurate network and customer service inventory information.

Manage new infrastructure requirements that IP Services necessitates. A single automated resource management platform is key to meeting the design/assign requirements in multiple network domains to enable support for multiple services. Manage new Resource Management paradigm that IP Services demands

Eliminate stovepipe with an end-to-end solution. A mechanism needs to be provided for consistency by normalizing the details and specifics of each service and related equipment. End customer experience is at stake. Efficient and effective Resource Management is the critical enabler to guarantee customer stickiness.

Reduces time, cost, and errors. Some methods can be used to reach this goal like

- Eliminates the need to configure each vendor's equipment independently
- Reduces errors by validating configurations prior to instantiation in the routers
- Enables rapid deployment of new customer services
- Dynamically configures "on-demand" or "scheduled" services
- Support configurations of value added services - VLANs, VPN, VoIP, IPTV

Quickly deploy new customer services. Dynamically configure new services with off-the-shelf models and flows for value added services are a must to enhance efficiency and customer's satisfaction.

6 End-to-End Service Assurance of China Mobile Solution

China Mobile has an End-to-End Service Assurance solution – it is included by One OSS system³.

One OSS can get data from Signaling system, auto-test system, IP backbone Network NMS, and the probes from NGN network. On the basis of standards we developed CMCC service model to cover all kinds of Network and VAS in service layer. And then through 10086 customer support system, one key sue & restoration system and customer's terminal support system (it can corrects the configuration errors on terminal).

It can be summed up Keys for Success in Building One-OSS as follows.

The major is Leverage Experience of Other Carriers Around the World, It Apply TM Forum initiatives including e-TOM, etc and learn from IP experiences of other operator. Secondly, It support Multi-Service Delivery on the IP network by integrate the legacy pipe stove system to support network cross service delivery and develop model for Multi-Service on the IP network. Thirdly, It can support Network and Service Operations by setup end to end service model for the services on IP network and enhance the network equipment lifecycle management, also increase the network efficiency. Fourthly, It Guarantee the QoS of network by develop the KPI for the IP network services and SQM function for QoS assurance. At last, It extend focus to business services by develop Customer Experience Management capabilities to improve customer satisfaction.

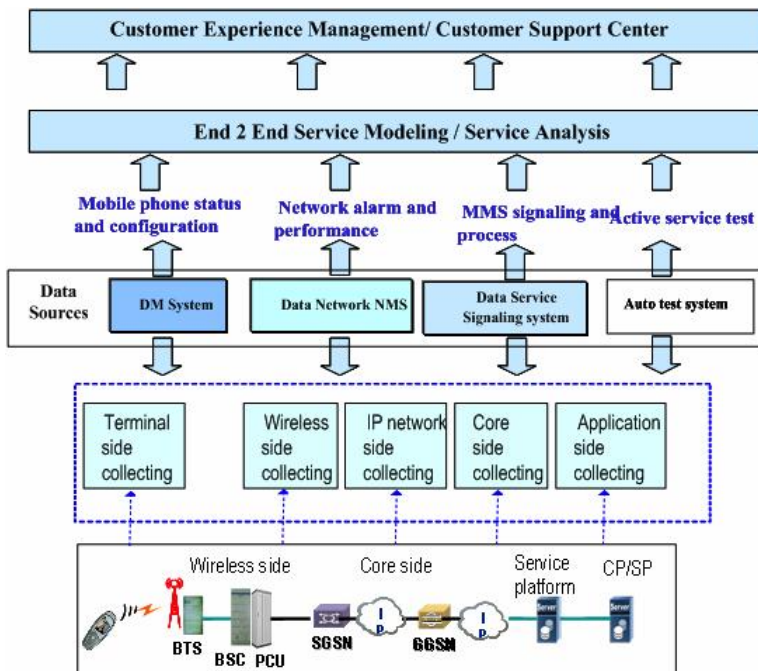


Fig. 2. End-to-End Service Assurance Model of China Mobile solution

Familiar as IP technologies are, systematic IP restructuring is new to the industry. It is a system engineer presenting tremendous challenges to achieve the IP transformation in the gigantic mobile networks. For this reason, endeavors must be made to follow the principles described below, which are of success and failure importance when we design OAM system.

When arduously setting up the operable and manageable IP network, take into consideration the deployment cycle of services and the progressing pace of supportive means. The deployment of new services in the IP network is now being accomplished at an increasingly fast speed, and challenges also stem from the lack of experience and means in managing traditional IP networks and IP services. For all these reasons, external supportive means and service management paces should be improved ASAP to ensure the capability for firmly controlling network elements and supporting new services. Following the above principle are the requisites for a smooth and successful IP transformation of the mobile networks.

7 Economic Returns of Mobile IP Networks

The successful IP transformation will definitely promote the fast provision of new services, lower the cost, and eventually increase the economic returns. Higher economic returns can be expected because with the improvement of bandwidth, cell phones are enabled to provide such services as locating, map navigating, and broadband games which used to be provided only by PCs. Besides, multi-media services such as mobile TV and enhanced voice services also join to represent the new growth points, which will make a breakthrough in the deadlock of increasing service volumes with decreasing economic benefits. On the other hand, lower costs can also be achieved. After the IP transformation, the local exchange of local traffic avoids the detour and decreases the signaling connections, which saves a lot of the

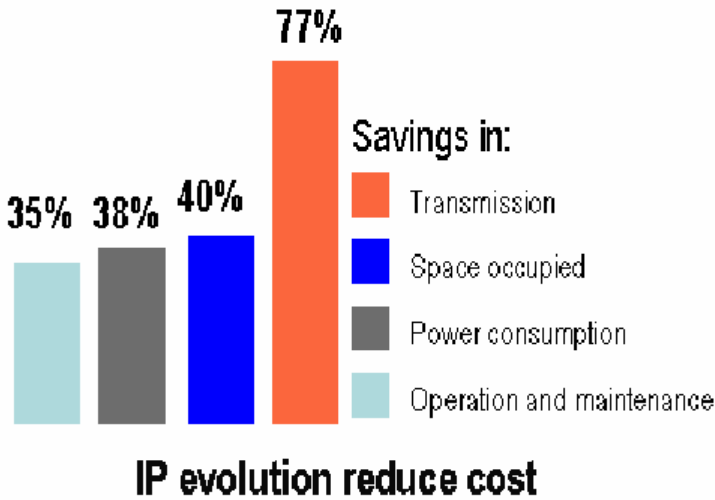


Fig. 3. Cost saving of IP evolution network to traditional network

transmission required originally. Besides, the adoption of a large amount of high integrated hardware also saves much electricity power, as well as spaces originally occupied by large size equipments. Large capacity equipments with fewer nodes allow central maintenance; the common open platform makes it possible for 2G and 3G networks to use the same nodes. All these contribute to cost saving. According to the industry research statistics, the mere replacement of MSC with a softswitch platform will reduce 50% of the OPEX.

Due to the above mentioned superiorities, some Chinese telecom operators who were not granted 3G licenses have even started drawing up an IP transformation blueprint for 2G networks, in an effort to save the cost for core network capacity expansion, prepare for future network multiplexing and gain advantages in competition.

The transformation to IP networks is the direction to which networks will progress in future, and also operators' only choice during network upgrades. It is an E2E process starting from core networks to access networks. The comprehensive IP restructuring, cost saving oriented, is undertaken by global leading mobile operators. Presently, around 90% of China Mobile's provincial tandem traffic volumes are connected through IP networks, which have demonstrated stable performance, as well as significant saving of cost and increase of benefit. We are convinced that in the near future, the All-IP mobile network will create more ecological chains and bring abundant multi-media experiences.

References

- [1] Lingshan, K.: IP evolution of Mobile network. *J. Communications world* 48 (July 2007)
- [2] Lihong, W., Lingshan, K.: The Roadmap of Mobile IP Network. In: *Proceedings of IEEE ICC 2008 China Forum*, pp. 26–29 (2008)
- [3] Lihong, W.: Take the challenge of All IP, *Telecom Management Forum, Focus on China, France Nice meeting*, May 18 (2008)

Performance Evaluation of Heartbeat-Style Failure Detector over Proactive and Reactive Routing Protocols for Mobile Ad Hoc Network

Haijun Zhao¹, Yan Ma^{1,2}, Xiaohong Huang¹, and Fang Zhao¹

¹ School of Computer Science and Technology
Beijing University of Posts and Telecommunications
Beijing, 100876, China

² Information Network Center
Beijing University of Posts and Telecommunications
Beijing, 100876, China

Abstract. Ad hoc wireless networks are characterized with multi-hop connectivity and frequently changing network topology. Routing protocol plays a very important role to ensure each packet reaches its destination node in Ad hoc wireless network. Many different routing protocols have been proposed in literature, but no performance comparison of heartbeat-style failure detection service over them is available. In this paper, we do the comprehensive simulations to investigate the performance of heartbeat-style failure detector over proactive and reactive routing protocols with ns-2 simulator. The performance different are analyzed using varying the number of nodes, mobility speed, transmission range. Results show that DSDV performed better in average delay and average failure detection time, AODV performed better in normalized routing overhead, delivery ratio, energy consumption ratio, false detection ratio. The observations can be used to motivate and improve future implementations of failure detection service in Ad hoc wireless networks.

Keywords: Ad Hoc Network, Failure Detection, Heartbeat, Performance.

1 Introduction

Failure detection service was a well-known fundamental building block for ensuring fault tolerance in distributed systems [1]. It has received much attention in the literature and has a wide application in network communication protocols [2], computer cluster management [3], group membership protocols [4] etc. Many heartbeat-style failure detection protocols have been proposed and implemented [6-10], however these approaches were either designed for static infrastructure or wired network and none evaluate its performance in Mobile Ad-Hoc wireless network.

Mobile Ad-Hoc network (MANET) [5] requires mobile nodes not only to be sources or destinations of packets but also to forward packets between other nodes. In other words, mobile node must act as routers. Routing protocol plays a very important role to ensure each packet reaches its destination node. Several published performance

studies of MANET routing protocols [11-13] have emphasized or focused on comparing or evaluating protocols based on constant bit rate (CBR) and TCP traffic. In this paper, we study performance of failure detection service over two typical Ad hoc wireless network routing protocols, DSDV from proactive approach [14] and AODV from reactive approach [15] with ns-2 simulator [16] by varying the number of nodes, mobility speed, traffic load, transmission range. The remainder of this paper is organized as follow. Section 2 introduces system model and basic failure detection. Section 3 introduces multi-hop routing in Ad Hoc network and performance metrics. Section 4 presents performance analysis. Section 5 gives simulation scenarios and results. Section 6 presents conclusion and future work.

2 System Model and Basic Failure Detection Algorithm

MANET is composed of a set of nodes which are placed in some physical domain, and each node may randomly move in this domain. Nodes can send heartbeat messages with some known transmission range, therefore nodes are connected by an unreliable fair lossy communication channel. For simplicity, we assume that all node clocks are synchronized with network time protocol.

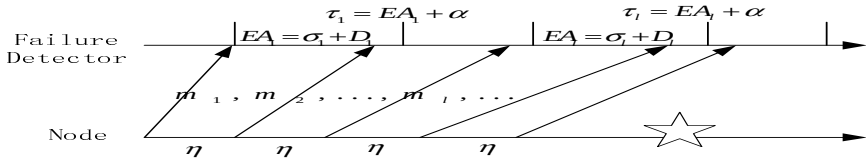


Fig. 1. Failure Detection Based on Timeout

The basic failure detection algorithm is generally based on heartbeat message timeouts (Fig 1). The monitored node periodically (every η interval) sends heartbeat messages $m_1, \dots, m_l, \dots$ to failure detector . The failure detector uses fixed time points $\tau_1, \dots, \tau_l \dots$ ($\tau_l = EA_l + \alpha$, $EA_l = \sigma_l + D_l$) and the time it receives heartbeat message to determine whether node is trusted or suspected. $t \in [\tau_l, \tau_{l+1})$, failure detector trusts node at time t if and only if failure detector has received heartbeat m_l or higher from this monitored node. EA_l denotes expected arrival time of the l^{th} heartbeat message, α denotes constant **safety margins**, σ_l denotes **send time of the l^{th} heartbeat**, D_l denotes **delay of the l^{th} heartbeat**.

3 Routing in Ad Hoc Networks and Performance Metrics

The proactive routing is derived from the traditional distance vector and link state protocols developed for use in the wire line Internet. The primary characteristic of

proactive approaches is that each node in the network maintains a route to every other node in the network at all times. Route creation and maintenance is accomplished through some combination of periodic and event-triggered routing updates. Periodic updates consist of routing information exchanges between nodes at set time intervals. The updates occur at specific intervals, regardless of the mobility and traffic characteristics of the network. Event-triggered updates, on the other hand, are transmitted whenever some event, such as a link addition or removal, occurs. The Destination-Sequenced Distance Vector (DSDV) routing protocol is a typical proactive routing, it utilizes per-node sequence numbers to avoid the counting to infinity problem common in many distance vector protocols. A node always selects the route with the greatest destination sequence number, this ensures utilization of the most recent information. DSDV utilizes both periodic and event-triggered routing table updates.

Reactive routing also called on-demand routing. Because link connectivity can change frequently and control overhead is costly in MANET, reactive routing approaches take a departure from traditional Internet routing approaches by not continuously maintaining a route between all pairs of network nodes. Instead, routes are only discovered when they are actually needed. When a source node needs to send data packets to some destination, it checks its route table to determine whether it has a route. If no route exists, it performs a route discovery procedure to find a path to the destination. Hence, route discovery becomes on-demand. Ad Hoc on Demand Distance Vector routing (AODV) is a typical reactive routing. Nodes maintain a route table which contains next-hop routing information for destination nodes. Each routing table entry has an associated lifetime value. If a route is not utilized within the lifetime period, the route is expired. AODV uses the destination sequence number to prevent routing loops and to determine freshness of routing information and nodes maintain only the route information for a destination corresponding to the latest known sequence number. The route discovery is based on flooding a route request (RREQ) and route reply (RREP) message. Once a route is established, it must be maintained as long as it is needed. Since the mobility of the nodes, links along paths are likely to break. In order to compare and evaluate the performance of failure detector over Ad hoc wireless routing protocols, we propose below metrics:

- **Normalized Routing Overhead:** Routing overhead consumes bandwidth, processing resources, and battery power to both transmit and receive a message. It is the number of routing packets transmitted per data packet sent to the destination. Also each forward packet is counted as on transmission.
- **Energy consumption ratio:** Most nodes are battery powered, and maximum energy efficiency is required for every task performed, including system maintenance, transmission, and reception of data. It is measured by energy consumed ratio to initial energy.
- **Delivery ratio:** It represents the fraction of data traffic that is successfully received at the intended destination, relative to the total amount of sent data.
- **Average Delay:** It is number of milliseconds elapsing from a heartbeat message is sent from the source and until it is received by failure detector. It includes all possible delays caused by buffering, during route discovery latency, processing, queuing at the MAC, and propagation and transfer times.

- **False failure detection ratio:** It represents the fraction of the number of false suspect relative to the total amount of decision.
- **Average Failure detection time:** It measures how fast it detects actual failures.

4 Performance Analysis

The main factors influence the above metrics are: the route path length to destination (the number of nodes), the rate of topological changes (mobility speed), transmission range etc.

Normalized routing overhead and energy consumption ratio depend on route path length, the route path stability is primarily determined by number of node and relative node mobility; Proactive protocols maintaining routes to every possible destination incurs a lot of unnecessary overhead, the periodic route update of proactive route results in a higher value for the normalized routing overhead. Higher normalized routing overhead will lead to higher energy consumption ratio. Reactive routing is naturally adaptive to topology change frequently and avoids incurring the cost of maintaining routes that are not used. Results in a lower normalized routing overhead and lower energy consumption ratio. Mobility does not alter this advantage of on-demand routing. Delivery ratio and false detection ratio are directly related to the packet drops. Packet drops typically happen because of network congestion (e.g., buffer overflows) or for lack of a route. Proactive protocols suffer from poor responsiveness to topology changes and slow convergence to optimal paths, which led to drop more heartbeat messages and results in lower delivery ratio and higher false detection ratio. However, the benefit is that when a route is needed, it is immediately available. Therefore, proactive protocols can obtain better delay and failure detection time. Reactive protocols can quickly generate new route path on-demand, which will lead to more heartbeat messages successfully delivered to destination and results in higher delivery ratio and lower false detection ratio. However, the disadvantage is that when a route is needed, it has to wait route discovery latency. Therefore, reactive protocols will bring longer delay and failure detection time. End-to-end delay and failure detection time depend on route discovery latency, additional delays at each hop (comprising of queuing, channel access and transmission delays), and the number of hops. As described in section II, since timeout values of failure detector depends on messages delay and the time it receives heartbeat message to determine whether monitored node is trusted or suspected. Longer delay increase the chances for a heartbeat to be received on time and as result decreases false detection ratio. But longer delay means longer failure detection time, because our algorithm use $1.2 \times \text{delay}$ as timeout, average failure detection time is reverse proportion to false detection ratio.

5 Simulation and Results

We evaluated the performance of basic failure detection algorithm over different routing protocol using ns-2 simulation [26]. The Monarch research group in CMU developed support for simulating multi-hop wireless networks complete with physical, data link and MAC layer models on ns-2. The distributed coordination function (DCF)

of IEEE 802.11 [5] for wireless LANs is used as the MAC layer. The 802.11 DCF uses Request-to-send (RTS) and Clear-to-send (CTS) control packets for “unicast” data transmission to a neighboring node. A network topology showed in Fig.2 is used for simulations and simulations are conducted by a set of basic parameters which are showed in Table 1. In our simulations we choose to study impact of changing one of these parameters while keeping the remaining. Thus in the following subsections, we will test the performance under below scenarios: varying the number of nodes (topology), varying node speed (mobility), varying transmission range.

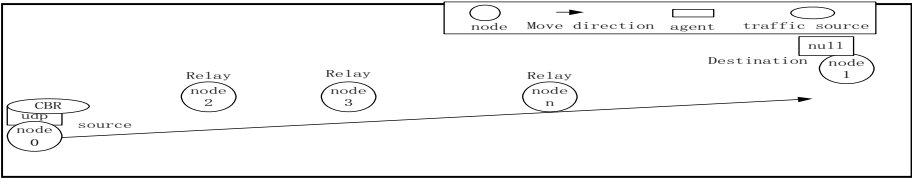


Fig. 2. Network Topology

Table 1. Basic simulation parameters

Parameter	Value
Tools	NS-2 with CMU extensions
Field size	1600X300 m ²
Channel/	WirelessChannel/ TwoRayGround/ OmniAntenna
MAC	IEEE 802.11/ DropTail/PriQueue
Initial Energy	1000mw
Bandwidth	2M
Heartbeat Message	Constant Bit rate (CBR) 100 bytes
Simulation time	150s
Source node	Node0 (monitored node). Initial position at (0,130). move to (1500,200)
Destination node	Node1 (Stationary failure detector) , at position (1600,300)
Number of nodes	Varying from 10 to 50 (Stationary relay nodes). at position $(i*(1600/n)+100,200)$
Node speed	Varying from 5 to 25 m/s
Traffic load	Varying from 0.01 to 0.05 s (message interval)
Transmission Range	Varying from 200 to 600m

● Performance of failure detector vs. number of Nodes

The number of nodes varies from 10 to 50, the effective transmission range of each node is 200 meters. node0 move (monitored node) at speed 25m/s. Node0 sends heartbeat message (CBR) of size 100 byte each 0.01 seconds. Results shown in Fig. 3 and Table2, the normalized routing overhead of AODV increases as the number of nodes increases. But it begin to fluctuates after a certain point (where the number of nodes 30), this is mainly due to nodes will be flooding the network with more RREQs to find routing. But the normalized routing overhead of DSDV drastically increases as the number of nodes increase, this is because more extensive and regular table updates are being sent. Note that with the same network density, AODV shows a smaller normalized routing overhead than DSDV. Both packet delivery rate of AODV and DSDV will increase with the number of nodes increasing. This is because network density increases will bring better network connectivity and decrease packets drop. But as the network density increase, there is an increase of radio interferences and collisions between nodes due to hidden/exposed terminals, therefore ratio begin fluctuates after a certain point (where the number of nodes 40 nodes (DSDV) or 30 nodes (AODV). Note that with the same network density, AODV shows a better

packet delivery rate than DSDV. The average delay of DSDV is pretty stable and keeps at 0.0019 second. But for on demand-driven AODV, it is hard to say the performance relationship, the curve jumped with number of nodes change. Note that with the same network density, DSDV shows a shorter average delay than AODV. The average energy consumption ratio of DSDV is pretty stable and keeps at 0.119. But for on demand-driven AODV, it is hard to say the performance relationship, the curve jumped with number of nodes change. Note that with the same network density, DSDV shows higher average energy consumption than AOD. Nodes increase will cause smaller false detection ratio for both DSDV and AODV. This is because more nodes improve the connectivity of the network and hence less heartbeat messages are lost. Note that with the same network density, AODV shows smaller false failure uetection ratio than DSDV. As expected, average failure detection time is reverse proportion to false detection ratio. For DSDV, Average failure detection time keep at 0.002 second and almost no change as increasing of the number of nodes, but for on demand-driven AODV, it is hard to say the performance relationship, the curve jumped with number of nodes change. Note that with the same network density, DSDV shows shorter average failure detection time than AODV.

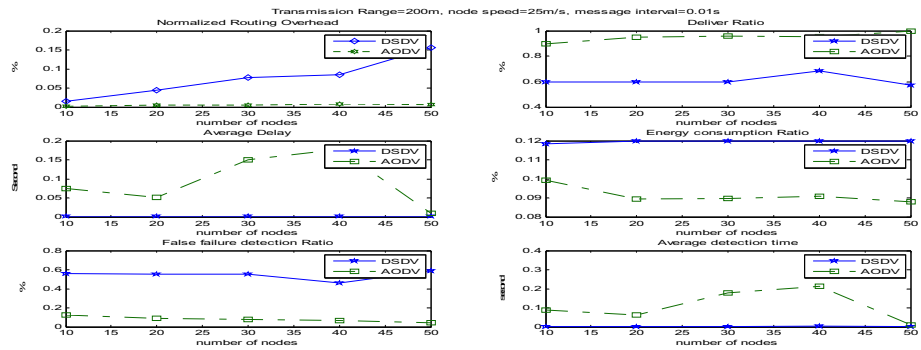


Fig. 3. Performance vs Number of nodes: node speed=25, Transmission Range=200m, Message interval=0.01s, simulation time=150s

Table 2. Performance vs Number of nodes

Number of nodes	10	20	30	40	50
Normalized Routing Overhead (DSDV)	0.015903	0.043692	0.078085	0.084319	0.155609
Normalized Routing Overhead (AODV)	0.002404	0.005413	0.004772	0.008189	0.006183
Deliver ratio(DSDV)	0.595750	0.598833	0.598917	0.683250	0.569583
Deliver ratio(AODV)	0.898000	0.949917	0.958000	0.953083	0.998917
Average Delay(DSDV)	0.001959	0.002020	0.001971	0.002530	0.001920
Average Delay(AODV)	0.074830	0.051558	0.149552	0.177775	0.009429
Energy consumption(DSDV)	0.118507	0.119844	0.119930	0.119824	0.119990
Energy consumption(AODV)	0.099266	0.089381	0.089662	0.090831	0.087965
False failure detection(DSDV)	0.558656	0.553386	0.556342	0.459555	0.590598
False failure detection(AODV)	0.125476	0.092013	0.081054	0.067287	0.040873
Average detection time(DSDV)	0.0023508	0.002424	0.0023652	0.003036	0.002304
Average detection time(AODV)	0.0089796	0.0618696	0.1794624	0.21333	0.0113148

● Performance of failure Detector vs. Speed of Nodes

The number of nodes is 10, the effective transmission range of each node is 200 meters, node0 (monitored node) move at speed varies from 5m/s to 25m/s. Node0 sends heartbeat message (CBR) of size 100 byte each 0.01 seconds. Results shown in Fig.4 and Table.3, Not surprising due to the extensive and regular updates of the

routing tables at the nodes, normalized routing overhead of DSDV drastically increases as speed of nodes increases. But AODV first increase then decreases, we notice that it tends to increase and saturate after a certain point (where the node speed is greater than 20m/s). The reason is speed increasing will improve connectivity of the network and not need generate more RREQs, but route will not be stable when speed greater than 20m/s, source node will have to generate more RREQs to find a new fresh enough route to destination. Note that with the same speed, AODV shows a smaller normalized routing overhead than DSDV. Both packet delivery rate of AODV and DSDV will increase with the speed of nodes increasing. This is because network density increases will bring better network connectivity.

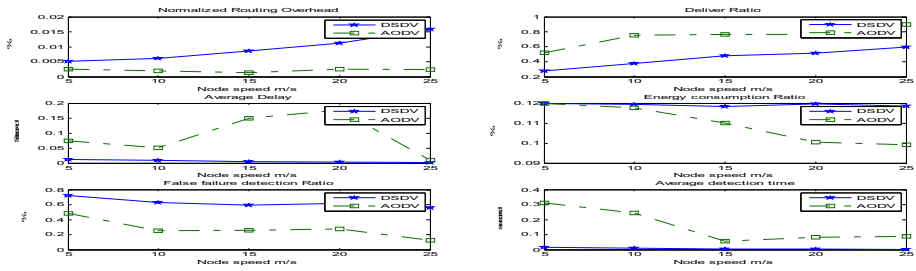


Fig. 4. Performance vs Node speed: number of nodes=30, Transmission Range=200m, Message interval=0.01s, simulation time=150 s

Table 3. Performance vs Node Speed

Node speed	5m/s	10m/s	15m/s	20m/s	25m/s
Normalized Routing Overhead (DSDV)	0.005116	0.006198	0.008563	0.011216	0.015903
Normalized Routing Overhead (AODV)	0.002550	0.001890	0.001331	0.002521	0.002404
Deliver ratio(DSDV)	0.277250	0.376583	0.483000	0.518333	0.595750
Deliver ratio(AODV)	0.519500	0.754667	0.767167	0.764917	0.898000
Average Delay(DSDV)	0.012740	0.008787	0.004735	0.003478	0.001959
Average Delay(AODV)	0.258600	0.204789	0.048062	0.068732	0.074830
Energy consumption(DSDV)	0.119999	0.119412	0.118314	0.119629	0.118507
Energy consumption(AODV)	0.119999	0.117702	0.110129	0.100601	0.099266
False failure detection(DSDV)	0.724553	0.627620	0.595661	0.615889	0.558656
False failure detection(AODV)	0.484349	0.250081	0.260638	0.274299	0.125476
Average detection time (DSDV)	0.015288	0.0105444	0.005682	0.0041736	0.0023508
Average detection time(AODV)	0.31032	0.2457468	0.0576744	0.0824784	0.089796

Note that with the same network density, AODV shows a better packet delivery rate than DSDV. Both average delay of AODV and DSDV will decrease with the speed of nodes increasing. The reason is that faster speed will cause shorter average routing path to destination. However the average delay of AODV will increase with speed 10m/s-20m/s, this is because the increase of speed induces topology change frequently and therefore the probability of broken links also grows, broken links may cause additional route recovery process and discovery process. Note that with the same speed, DSDV shows a shorter average delay than AODV. The average energy consumption ratio of DSDV is pretty stable and keeps at 0.119 and almost no change as speed increase. The reason is that DSDV is a table-driven, so a node does not need to find a route before transmitting packets. But average energy consumption ratio of DSDV will decrease, the increase of speed will trigger node to find shorter routing path to destination and save more energy. Note that with the same speed, DSDV

shows higher average energy consumption than AODV. Node speed increase will cause smaller false detection ratio for both DSDV and AODV. This is because increase of speed will also improve the connectivity of the network and hence less heartbeat messages are dropped. Note that with the same speed, AODV shows smaller false failure detection ratio than DSDV. As expected, average failure detection time is reverse proportion to false detection ratio. Longer failure detection time allow heartbeat message routing along longer paths and as a result increase the chances for a heartbeat to be received on time. Since our algorithm use $1.2 \times \text{delay}$ as timeout, so average failure detection time behaves similar as the average delay.

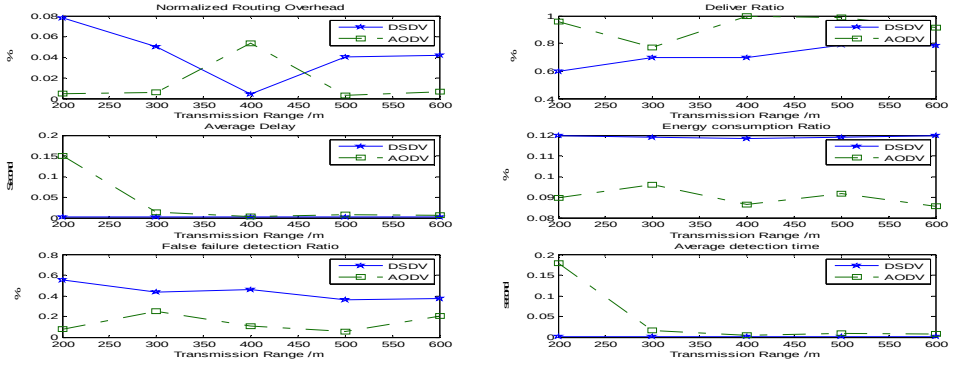


Fig. 5. Performance vs Transmission range: number of nodes=30, node speed=25m/s, message interval=0.01s, simulation time=150s

Table 4. Performance vs Transmission range

Transmission Range	200m	300m	400m	500m	600m
Normalized Routing Overhead (DSDV)	0.078085	0.050375	0.004356	0.040368	0.042139
Normalized Routing Overhead (AODV)	0.004772	0.006294	0.053783	0.003036	0.006821
Deliver ratio(DSDV)	0.598917	0.699000	0.699333	0.789000	0.783667
Deliver ratio(AODV)	0.958000	0.770917	0.999083	0.989667	0.914667
Average Delay(DSDV)	0.001971	0.002181	0.001760	0.001977	0.001779
Average Delay(AODV)	0.149552	0.013912	0.004090	0.007892	0.007160
Energy consumption(DSDV)	0.119930	0.118905	0.118312	0.119025	0.119859
Energy consumption(AODV)	0.089662	0.096131	0.086611	0.091717	0.085756
False failure detection(DSDV)	0.556342	0.439038	0.459013	0.362442	0.373909
False failure detection(AODV)	0.081054	0.253727	0.107088	0.056810	0.205676
Average detection time(DSDV)	0.0023652	0.0026172	0.002112	0.0023724	0.0021348
Average detection time (AODV)	0.1794624	0.0166944	0.004908	0.0094704	0.008592

● Performance of Failure Detector vs. Transmission Range

The number of nodes is 30, the effective transmission range of each node varies from 200 to 600 meters. Node0 (monitored node) move at speed 25m/s. Node0 sends heartbeat message (CBR) of size 100 byte at interval 0.01 seconds. Results shown in Fig. 5 and Table.4, normalized routing overhead of DSDV first decreases then increases again. This is because transmission range increases from 200m to 400m will improve network connectivity. But when transmission range changes from 400 to 600m, there is an increase of radio interferences and collisions between nodes due to hidden/exposed terminals. AODV show reverse behavior, this is because topology change frequently also trigger AODV to generate more route discovery overhead to find new route, when transmission range exceeds 400m, the normalized routing overhead

of AODV will fluctuate. Note that except transmission range from 350m to 450m, AODV shows a smaller normalized routing overhead than DSDV, because on-demand routing protocols can reflect topology change in route information right away, but the table-driven protocol maintains exhaustive information about the network topology. The packet delivery ratio of DSDV will increase as transmission range increases. This is because the more transmission range, the better network connectivity. It is hard to say the performance relationship between AODV and transmission range, the curve jumped with transmission range changes. This is because when transmission range changes from 200 to 300m, there is an increase of radio interferences and collisions between nodes due to hidden/exposed terminals. Transmission range change from 300 to 400m will improve network connectivity. But when changes from 400 to 600m, radio interferences and collisions increase again. Note that with the same transmission range, AODV shows a better packet delivery rate than DSDV as before. The average delay of DSDV is pretty stable and keeps at 0.0019 second, the reason is that DSDV is a table-driven, so a node does not need to find a route before transmitting packets. But the average delay of AODV decreases when transmission range increases from 200 to 400m, this is because the better network connectivity, the shorter hops to destination. When transmission range exceeds 400m, the average delay of AODV will fluctuate again. Note that with the same transmission Range, DSDV shows a shorter average delay than AODV as before. The average energy consumption ratio of both DSDV and AODV all fluctuate. The reason is that the influence of transmission range increases, topology changes frequently, radio interferences and collisions, which cause nodes sometime to consume more energy, sometime to save energy. Note that with same transmission range, DSDV shows higher average energy consumption than AODV as before. The false detection ratio of DSDV will decrease as transmission range increases. The reason is that DSDV can keep stable average delay and increase packet delivery rate. The false detection ratio of AODV will fluctuate since packet delivery rate and average delay will fluctuate as transmission range increases. Note that with the same transmission range, AODV shows a better false detection ratio than DSDV as before. As expected, average failure detection time is reverse proportion to false detection ratio.

6 Conclusion

In order to overcome the barrier for MANET to widely use, there is a need for middleware services that support ubiquitous applications. Failure detection is a fundamental middleware in many reliable distributed services and applications. A variety of routing protocols have been proposed for MANET, however none focus on study the performance of heartbeat-style failure detector over them. We evaluate such performance over proactive and reactive routing protocols with ns-2 simulator by varying the number of nodes, node speed, transmission range. The performance metrics that we considered include normalized routing overhead, packet delivery ratio, average delay, average energy consumption ratio, false detection ratio, average failure detection time. Numerical and graphical results showed normalized routing overhead and average energy consumption ratio is tightly coupled with the stability and length of routing paths. Delivery ratio and false detection ratio are directly related

to the packet drops. Average delay and failure detection time depend on route discovery latency, additional delays at each hop and the number of hops. We plan to investigate the performance of heartbeat-style failure detector over peer to peer network.

Acknowledgments. This work is supported by China National Key Foundation Research 973 project (No.2003CB314806) and National Natural Science Foundation of China (60772111).

References

1. Chandra, T.D., Toueg, S.: Unreliable failure detectors for reliable distributed systems. *Journal of the ACM* 43(2), 225–267 (1996)
2. Braden, R. (ed.): Requirements for Internet Hosts-Communication Layers, RFC 1122 (October 1989)
3. Pfister, G.F.: Search of Clusters, 2nd edn. Prentice Hall, Englewood Cliffs (1998)
4. van Renesse, R., Birman, K.P., Maffei, S.: Horus: A Flexible Group Communication system. *Comm. ACM* 39(4), 76–83 (1996)
5. National Science Foundation. Research priorities in wireless and mobile communications and networking: Report of a workshop held March 24–26, Airlie House, Virginia (1997), <http://www.cise.nsf.gov/anir/www.html>
6. Chen, W., Toueg, S., Aguilera, M.K.: On the quality of service of failure detectors. *[J].IEEE Transactionson Computers* 51(2), 13–32 (2002)
7. Bertier, M., Marin, O., Sens, P.: Implementation and performance evaluation of an adaptable failure detector. In: Martin, D.C. (ed.) *Proc. of the 15th Int'l Conf. on Dependable Systems and Networks*, Bethesda, pp. 354–363. IEEE CS Press, Los Alamitos (2002)
8. Falai, L., Bonvadalli, A.: Experimental evaluation of the QoS of failure detectors on wide area network. In: Tsuchiya, T. (ed.) *Proc. of the Int'l Conf. on Dependable Systems and Networks (DSN 2005)*, Yokohama, pp. 624–633. IEEE CS Press, Los Alamitos (2005)
9. Chen, N.J., Wei, J., Yang, B., Huang, T.: Adaptive failure detection in Web application server. *Journal of Software* 16(11), 1929–1938 (2005)
10. Défago, X., Urbán, P., Hayashibara, N., Katayama, T.: On accrual failure detectors. RR IS-RR-2004-11, JAIST,Ishikawa, Japan (May 2004)
11. Dyer, T.D., Boppana, R.V.: A comparison of TCP performance over three routing protocols for mobile ad hoc networks. In: *Proceedings of ACM Mobihoc*, pp. 56–66 (October 2001)
12. Das, S.R., Perkins, C.E., Royer, E.M.: Performance comparison of two on-demand routing protocols for ad hoc networks. In: *IEEE Infocom 2000* (March 2000)
13. Johansson, P., et al.: Scenario-based performance analysis of routing protocols for mobile ad-hoc networks. In: *ACM Mobicom 1999* (August 1999)
14. Perkins, C.E., Bhagwat, P.: Highly dynamic destination-sequenced distance vector (DSDV) for mobile computers. In: *ACM SIGCOMM 1994*, pp. 234–244 (August 1994)
15. Perkins, C., Royer, E.: Ad hoc on-demand distance vector routing. In: *Proceedings of the 2nd IEEEWorkshop on Mobile Computing Systems and Applications*, pp. 90–100 (February 1999)
16. CMU Monarch Group, CMU Monarch extensions to the NS-2 simulator (1998), <http://monarch.cs.cmu.edu/cmu-ns.html>

Design and Implementation of an SNMP-Based Traffic Flooding Attack Detection System

Jun-Sang Park and Myung-Sup Kim

Dept. of Computer and Information Science, Korea University, Korea
{runtoyou, tmskim}@korea.ac.kr

Abstract. Recently, as traffic flooding attacks such as DoS/DDoS and Internet Worm have posed devastating threats to network services, rapid detection and proper response mechanisms are the major concern for secure and reliable network services. However, most of the current Intrusion Detection Systems (IDSs) focus on detail analysis of packet data, which results in late detection and a high system burden to cope with high-speed network traffic. In this paper we propose an SNMP-based lightweight and fast detection algorithm for traffic flooding attacks, which minimizes the processing and network overhead of the detection system, minimizes the detection time, and provides high detection rate.

Keywords: Traffic Flooding Attack, DoS/DDoS, SNMP, MIB, Detection Algorithm, Detection Time, Detection System.

1 Introduction

Recently, network flooding attacks such as DoS/DDoS and Internet Worm have posed devastating threats to network services. Moore et al. [1] reported that the DoS/DDoS attack is the main threat to the entire Internet, and the majority of them (90-94%) are deployed by using TCP. As a result, rapid detection and fast response mechanisms are the major concern for secure and reliable network services [2].

Most of the current Intrusion Detection Systems (IDSs) investigate packet data to evaluate the security status of the network and system, which results in a significant processing burden and eventually, late detection time. Little or no integration exists between IDS and SNMP-based Network Management Systems (NMS) in spite of the extensive monitoring and statistical information offered by SNMP agents running on network elements. For example, SNMP MIB-II, an IETF standard MIB supported by all the SNMP agents, provides a large number of traffic performance information on different layers and protocols: IP, TCP, UDP, ICMP, etc. SNMP agents are already implemented in most current network elements, thus, MIB statistical data are available to easily collect for security analysis. This can be extended to collect additional data pertinent to network activities and is independent of the operating systems. By enlisting the MIB data, IDS promises a lower processing overhead for analysis, and high flexibility of deployment. [2, 3, 6, 7, 8]

In this paper, we propose a lightweight and fast detection algorithm for traffic flooding attack via correlations of SNMP MIB data.

The paper is organized as follows. In section 2 we describe related work and problems in the design of SNMP-based detection system. Section 3 presents the proposed traffic flooding attacks detection algorithm. Section 4 describes the design of the proposed system. Section 5 evaluates the validity of the proposed algorithm via experiment. Section 6 concludes this paper, with conclusions and future work.

2 Related Work

SNMP provides a universal method of exchanging data for purposes of monitoring systems that reside on a network. But, to utilize SNMP for traffic flooding attack detection we need to consider the following three points in the use of the SNMP MIB objects which affects the performance and accuracy of the detection system: 1) Proper selection of SNMP MIB objects for attack detection, 2) Determination of the detection timing about when and how often, 3) Algorithm for attack detection using the selected MIB objects.

The first, we need to select a proper SNMP MIB objects for attack detection. This selection should be done to meet that the number of SNMP MIB objects involved is minimized and the range of attack types covered is maximized. Yoo et al. [3] utilized tcpInErrs, udpNoPorts, and icmpOutEchoReps SNMP MIB objects for the purpose of detecting DoS/DDoS attacks. These MIB objects are not suitable to detect certain type of modern attacks. Attackers scan vulnerabilities of a victim host first before sending a large flood of packets targeting the victim host, which sabotage both systems resources and network resources. Therefore the MIB objects used in [3] may not contribute much as before on detecting modern attacks.

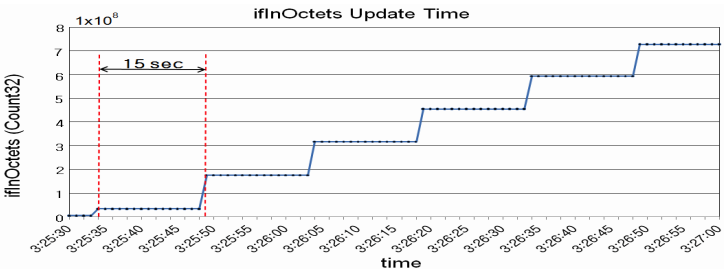


Fig. 1. Periodic update of ifInOctets MIB object

For the second consideration, we need to determine the detection timing: when and how often the detection system is triggered to retrieve SNMP MIB objects from a system and analyze them to decide the system is normal or abnormal. Short detection interval gives a fast detection while high processing and traffic overhead occurs as a side-effect, but long detection interval results in late detection with low burden in system and network. We have to determine a suitable detection time and interval.

Figure 1 illustrates the change of *ifInOctets* MIB value in the interval of 1 second which is gathered from a Net-SNMP SNMP agent running on a Linux. Figure 1 says that SNMP MIB variables are periodically updated with a certain time interval. The

ifInOctets MIB objects of Net-SNMP agent is updated in about every 15 seconds. We figured out that the update interval is different from system to system and from agent to agent. If we don't consider the periodic update of SNMP MIB values, the MIB values retrieved at a certain time is the values in 15 seconds before in the worst case. The best is to retrieve MIB values just after the MIB values are updated in the SNMP agent. By triggering detection system whenever the MIB variables are updated we can achieve the requirement of fast detection. But detection of update time is another problem to be solved. System load and network overhead must be considered in order to manage multiple target systems.

Thirdly, we have to develop an algorithm for attack detection using the SNMP MIB data gathered from a target system. This is the problem of how to utilize them for detection. Yoo et al. [3] constructed a normal trend of SNMP MIB data which is compared with the gathered MIB data. Cabrera et al. [6] also used SNMP MIB for symptom analysis of the DDoS attack using a statistical analysis. Machine learning based classification mechanism was used in [9]. These approaches detect traffic flooding attacks based on the learning of past normal and attack traffic data, which result in some amount of false negative rate (FNR) because of the difficulty in adapting to the dynamic nature of modern multilateral network traffic.

3 Traffic Flooding Attack Detection Algorithm

In this section, we present the proposed detection algorithm based on the correlation of SNMP MIB data, and means of determining the detection timing which minimized processing and traffic overhead.

We selected 16 MIB objects from 6 groups (system, interface, ip, tcp, udp, icmp) in mib-2 defined in RFC1213. [4] The data types of the 15 MIB objects except *sysUpTime* in Table 1 are Counter32, which is a non-negative 4-byte integer continuously incremented. These MIB variables are continuously updated as the outgoing and incoming traffic occur from/to a system. We selected 15 MIBs which responds to the attack traffic by a comprehensive investigation.

Table 1. The MIB used in attack detection

MIB-2 Group	SNMP MIB objects
system	system. <i>sysUpTime</i>
interface	interface.ifTable.ifEntry. <i>ifInOctets</i> interface.ifTable.ifEntry. <i>ifInUcastPkts</i>
ip	ip. <i>ipInReceives</i> ip. <i>ipInDelivers</i> ip. <i>ipOutRequests</i> ip. <i>ipOutDiscards</i>
tcp	tcp. <i>tcpAttemptFails</i> tcp. <i>tcpOutRsts</i>
udp	udp. <i>udpInErrors</i>
icmp	icmp. <i>icmpInMsgs</i> icmp. <i>icmpInDestUnreachs</i> icmp. <i>icmpInEchos</i> icmp. <i>icmpOutMsgs</i> icmp. <i>icmpOutDestUnreachs</i> icmp. <i>icmpOutEchoReps</i>

Table 2. The numerical equation and notation

t	Time, 1 second unit
t_n	The time of n -th detection algorithm worked
$mib(t_n, oid)$	The value of MIB oid at time t_n
$diff(t_n, oid)$	$= mib(t_n, oid) - mib(t_{n-1}, oid)$
$bps(t_n)$	$= 800 * diff(t_n, ifInOctets) / diff(t_n, sysUpTime)$
$pps(t_n)$	$= 100 * diff(t_n, ifInUcastPkts) / diff(t_n, sysUpTime)$
$DeliverRatio(t_n)$	$= diff(t_n, ipInDelivers) / diff(t_n, ipInReceives)$
$ResponseRatio(t_n)$	$= diff(t_n, ipOutRequests) / diff(t_n, ipInReceives)$

To simplify our discussion, Table 2 presents numerical equations and notations to express the detection algorithm.

Figure 2 presents a flowchart of the proposed traffic flooding attack detection algorithm. It consists of three consecutive stages; determining the detection timing according to the update interval of SNMP MIB (Stage 1), analyzing the symptom of attacks (Stage 2) and classifying the attack (Stage 3).

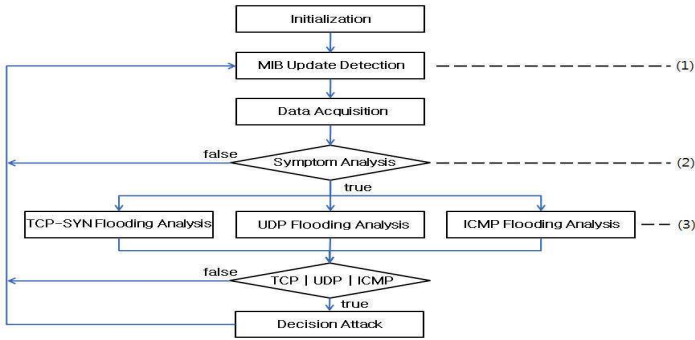


Fig. 2. Flow chart of detection algorithm

The first stage determines the time when MIB object is updated and triggers detection system with minimal overhead. In the second stage, based on interface and ip MIB objects the possibility of attack is investigated. If there is some possibility of attacks, the system performs a detail analysis for each attack type according to the transport layer protocols MIB data and determines the attack state as well as attack type. If there is no symptom in second stage, the system moves to the first stage and detects next MIB update. Subsequent to detecting the update of MIB variables, the MIB objects used for attack detection are collected from an SNMP agent in the target system via a pair of SNMP message (GetRequest and Response), which minimize the amount of network traffic.

3.1 Stage 1: Determination of Detection Timing

Stage 1 improves the detection time using an MIB update interval. The detection time is determined via the update time. Accordingly, we can achieve quick detection and obtain accurate variables from an SNMP agent.

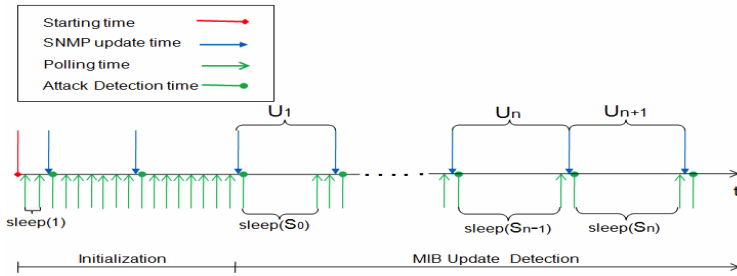


Fig. 3. Algorithm for determination of the detection time

Figure 3 depicts the algorithm that determines the detection time. In the initialization phase, the system collects *ifInOctets* MIB every 1 second, determines the MIB update, and executes the detection system when MIB data is updated. The minimum update interval ($U_{\min}$) and the time (t_{old}) at which attack detection was last applied are used in the subsequent stage as the initial values ($U_0 = U_{\min}$).

The MIB update detection phase predicts the next MIB update time for determination of the sleep time. The detection system does not operate prior to update of the *ifInOctets* value. Hence, the management system has reduced processing and traffic load. In order to determinate the detection time, we use the following equations, which calculate the exponential average (A_n) of period for updating MIB values. Constant value α is set to 0.5. Finally, the next sleep time (S_n) is calculated by the following equation.

$$A_n = \alpha \times U_n + (1 - \alpha) \times A_{n-1}$$
$$S_n = A_n - 1$$

3.2 Stage 2: Analyzing the Symptoms of the Attack

Symptom analysis of traffic flooding attacks is based on the amount of traffic and the delivering ratio at the target system, which distinguishes definitely normal states from possibly abnormal states. This reduces additional processing load at the detection system and true negative rate. Traffic flooding attacks consume much bandwidth and affect network performance. Therefore, if the attacks reach the target system, the parameters related on bandwidth associated with the target system is affected, such as bps (bits per second) and pps (packets per second). The result determines whether an additional analysis is executed or not. Then, identification of the symptoms of attacks is based on *DeliverRatio()*, *ResponseRatio()* and *diff()*, which represent correlations of the ip group MIB objects. The algorithm analyzing symptoms of traffic flooding attacks is formalized as follows;

```
Boolean Symptom_analysis( ... ){
    int weight = 0;
    if(bps < Th(bps) && pps < Th(pps)) return FALSE;

    if( DeliverRatio(t) < Th(DeliverRatio) )           weight++;
    if( ResponseRatio(t) < Th(ResponseRatio) )         weight++;
    if( Diff(t, ipOutDiscards) > Th(ipOutDiscards) )    weight++;
    if(weight >= 1) return TRUE;
    return FALSE;}
}
```

`DeliverRatio()` represents the rate of packet transmission from IP layer to transport layer at the target system. Generally, this exceeds 80% during normal state. In case of flooding attack, it is less than 30% because the delivered packets have errors (checksum error) or the buffer capacity is insufficient.

`ResponseRatio()` indicates the number of packet responses that the target system sends to remote hosts. When the target system receives normal packets, it exceeds 50%. But, in the other case it is less than 40% because the response packets are dropped by lack of buffer capacity.

`ipOutDiscards` shows the number of dropped packets. This value is increased by lack of buffer capacity for transmission to spoofed hosts in case of flooding attack.

In this stage we use several constant thresholds values for determining whether the traffic is absolutely normal or possibly abnormal. The thresholds of BPS and PPS include the number of packets that an adversary can generate. The thresholds used in BPS and PPS must guarantee the reliability of `DeliverRatio()` and `ResponseRatio()`. We use static values which are determined by a comparison between normal traffic and attack traffic for `DeliverRatio()`, `ResponseRatio()` and `diff()`, which is shown in Table 3.

Table 3. The threshold values used in symptom analysis

Th(bps)	1 Mbps	Th(pps)	20 pps
Th(DeliverRatio)	0.8	Th(ResponseRatio)	0.4
Th(ipOutDiscards)	0		

3.3 Stage 3: Decision Attack Based Protocol Analysis

Protocol analysis modules detect each type of attack using tcp, udp and icmp MIB groups, since there are one or more MIBs that substantially monitor each attack. In this stage, each module detects an attack via traffic variations exceeding the threshold, then, classifies the attack.

Detection of TCP SYN Flooding Attacks

The `tcpAttamptFail` MIB value is increased by connection failure between spoofed systems and target systems. The `tcpOutRsts` value increases if the corresponding port is closed in the target system during attack runs. The target system transmits packets which include the RST flag.

```

Boolean TCP-SYN_Flooding_analysis( ... ){
    if(Diff(t,tcpAttamptFail)>Th(tcpAttamptFail) || Diff(t, tcpOutRsts )> Th(tcpOutRsts) ) )
        return TRUE;
    return FALSE;}
    
```

Detection of UDP Flooding ATTACKS

The `udpInErrs` value is increased by errors and lack of buffer capacity. In addition, if packets are not delivered to the application layer, the target system sends an ICMP error reporting message to the spoofed IPs. Thus, the `icmpOutDestUnreachs` increases.

```
Boolean UDP_Flooding_analysis( ... ){
    int weight = 0;
    if( Diff(t, udpInErrs) > Th(udpInErrs))                weight ++;
    if(Diff(t, icmpOutDstUnreachs)==diff(t,icmpOutMsgs) &&
        Diff(t, icmpOutDstUnreachs) > 0 && Diff(t, icmpOutMsgs) > 0 &&
        Diff(t, icmpInDstUnreachs) == 0)                    weight ++;
    if(weight > =1) return TRUE;
    return FALSE;}
}
```

Detection of ICMP Flooding Attacks

The *icmpInEchos* and the *icmpInMsgs* are substantially increased. Then, the target system sends packets which correspond to echo request packets, and packets are dropped by buffer overflow. Therefore, the *ipOutDiscards* increases. If the attack packets are not delivered to the application layer, the target system sends an ICMP error reporting message to the spoofed IP. Thus, the *icmpInDestUnreachs*, *icmpOutDestUnreachs* values increase.

```
Boolean ICMP_Flooding_analysis( ... ){
    int weight=0;
    if( Diff(t, icmpInMsgs) > Th(icmpInMsgs) ||
        Diff(t, icmpInEchos) > Th(icmpInEchos) )                weight++;
    if( Diff(t, icmpInDstUnreachs) > 0 &&
        Diff(t, icmpOutDstUnreachs) > 0 &&
        Diff(t, icmpOutEchoReps) > 0 &&
        Diff(t, ipOutDiscards > 0) )                            weight++;
    if(weight > =1) return TRUE;
    return FALSE;}
}
```

4 Design and Implementation of Detection System

We designed and implemented a detection system based on the proposed algorithm. The detection results are reported to the network manager using a web interface. In addition, the system sends an SMS message for immediate reporting the occurrence of attacks to the network manager.

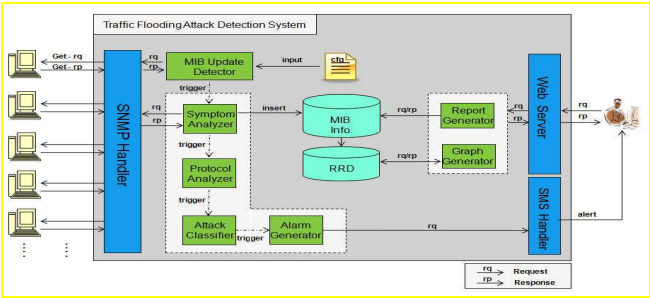


Fig. 4. Overall system architecture for the attack detection system

As illustrated in Figure 4, detection system tasks are partitioned into several modules. This provides good scalability and flexibility. The top layer of the system

consists of a Web server, which communicates with the administrators with HTTP, and an SMS Handler, which communicates with the administrator. In addition, there is an SNMP Handler at the other end, which communicates with a target system with SNMP. The middle layer of the system consists of; determination of the detection time, detection module, alarm generator, MIB info. DB, RRD, Graph generator, and report generator. The detection system allocates a process to each target system. Each process collects the SNMP MIB data required for detection, saves MIB data, and controls all detection stages. MIB data is provided and managed by each system. The MIB Info. DB is converted to an RRD DB to generate a time-series graph.

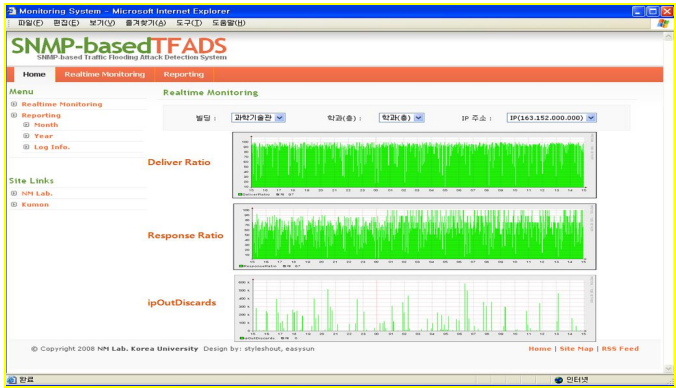


Fig. 5. Web user interface of the proposed detection system

The report generator shows the results for each target system communicating with the detection system, if the administrator desires web services. If the administrator requests information about the target system using a web server, the report generator shows the data stored in DB. This provides graphs of DeliverRatio(), ResponseRatio() and ipOutDiscards() using RRD. In addition, log data of an attack and statistical data are provided, if requested by the administrator. If an attack is detected, information including the target system, the attack time, and the attack type is provided to the administrator in real-time using SMS service.

5 Experiments

In this section we show the results of performance measurements, to certify the validity of the detection algorithm. The tests consisted of three attack systems, a target host and a management system. The SNMP agent at the target system updates MIB objects at 15 second intervals, on average. We used Stacheldraht[5] to perform TCP-SYN, UDP and ICMP flooding attacks. The tests were performed over a period of 10 days. But, during the experiment only a single attack was performed by attack system at a time. The type and duration for each attack was selected in a random manner. To describe the performance of the detection algorithm, we focus on the TCP-SYN flooding attack.

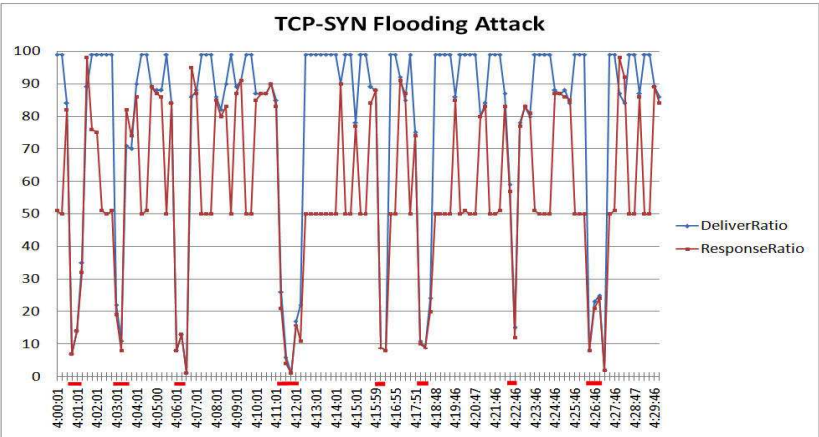


Fig. 6. Delivery rate and response rate at target system

Regarding Figure 6, it is remarkable that DeliverRatio and ResponseRatio are less than 0.8 and 0.4, respectively, during TCP-SYN flooding attacks. Since the detection algorithm analyzes symptoms of the attack. A TCP-SYN flooding attack is detected via *tcpAttemptFail* and *tcpOutRsts* in the protocol analysis module. Then, the attack is classified.

Detection of each attack required 8.3 seconds on average after attack starts. This result guarantees that the system has sufficient stability. The detection algorithm does not affect the management system, since it applies an exponential average to determine the sleep time and uses a hierarchical detection stages. In addition, there was negligible traffic load, due to SNMP messages used to determinate the detection time. Accordingly, we can implement a detection system which ensures scalability for managing multiple systems.

Table 4. Detection rates according to the number of attacks and detection

Attack Type	TCP-SYN	UDP	ICMP	Normal	Total
# attack	794	832	802	0	2428
TCP-SYN	100%	0%	0%	0%	794
UDP	0%	100%	0%	0%	832
ICMP	0%	0%	100%	0%	802
# detection	2526	2769	2613	49732	57640
TCP-SYN	97.34%	0%	0%	2.66%	2526
UDP	0%	99.2%	0%	0.8%	2769
ICMP	0%	0%	100%	0%	2613
Normal	0.03%	0.06%	0.9%	99.82%	49732

We proved that the proposed detection algorithm could precisely detect all attacks in 1-2 times of detection operation. In terms of detection operation, there was a few FP (False Positive) and FN (False Negative). All of the false detection occurs only when the difference between the attack start time and detection time was within 1 second. We believe this can be ignored because the attack detection has an error of less than 1 second in nature.

6 Conclusion

In this paper we proposed a lightweight and fast traffic flooding attack detection algorithm using SNMP and a method for improving the detection time. The proposed system consists of three consecutive stages: MIB update detection, Symptom Analysis, and Protocol Analysis. By predicting the next MIB data update time we can remove the unnecessary usage of SNMP message and system burden. In the symptom analysis by distinguishing definitely normal traffic state, which is most of the traffic state, from the possibly abnormal state, we could reduce the processing burden. By the protocol analysis we could accurately detect traffic flooding attack and determine the type of attacks. We also designed and developed the proposed traffic flooding attack detection system and validated the applicability of the proposed system by real-world experiment. In addition, the proposed algorithm provides good scalability and extendibility for managing a large number of target systems.

For a possible future work we are planning to scale our detection algorithm to network environment. Also we are trying to extend this system for the detection of attack agents in DDoS mechanism, which is more valuable and promising.

Acknowledgments. This work was supported by the Korea Research Foundation Grant funded by the Korean Government (MOEHRD, Basic Research Promotion Fund) (KRF-2007-331-D00387).

References

1. Moore, D., Voelker, G., Savage, S.: Inferring Internet Denial-of-Service Activity. In: Proc. of the Usenix Security Symposium, Washington, D.C, pp. 401–414 (2001)
2. Duarte Jr., E., dos Santos, A.L.: Network Fault Management Based on SNMP Agent Groups. In: Proc. of ICDCSW, p. 51 (2001)
3. Yoo, D.-S., Oh, C.-S.: Traffic Gathering and Analysis Algorithm for Attack Detection. In: KoCon 2004 Spring Integrated conference, vol. 4, pp. 33–43 (2004)
4. IETF RFC 1213. Management Information Base for Network Management of TCP/Ip-Based Internets: MIB-II, <http://www.rfc-editor.org/rfc/rfc1213.txt>
5. Distributed Denial of Service (DDoS) Attacks/tools, <http://staff.washington.edu>
6. Li, J.: Constantine Manikopoulos.: Early Statistical Anomaly Intrusion Detection of DoS Attacks Using MIB Traffic Parameters. In: Proc. of the IEEE WIA, pp. 53–59 (2003)
7. GasparyL, P., Sanchez, R.N., Antunes, D.W., Meneghetti, E.: A SNMP-based platform for distributed stateful intrusion detection in enterprise network. IEEE Journal on Selected Areas in Communications 23, 1973–1982 (2005)
8. Cabrera, J.B.D., Lewis, L., Qin, X., Lee, W., Prasanth, R.K., Ravichandran, B., Mehra, R.K.: Proactive detection of distributed denial of service attacks using MIB traffic variables-a feasibility study. In: IEEE/IFIP International Symposium, pp. 606–622 (2002)
9. Xue, Q., Guo, L.-L., Sun, J.-Z.: The design of a distributed network intrusion detection system IA-NIDS. In: International Conference on Machine Learning and Cybernetics 2003, vol. 4, pp. 2305–2308 (2003)
10. Carlsson, P., Fiedler, M., Tutschku, K., Chevul, S., Nilsson, A.A.: Obtaining Reliable Bit Rate measurements in SNMP-Managed Networks: ITC Specialist Seminar, Würzburg, pp. 114–123 (2002)

A Method to Detect Prefix Hijacking by Using Ping Tests

Mitsuho Tahara, Naoki Tateishi, Toshio Oimatsu, and Souhei Majima

NTT Network Service Systems Laboratories, NTT Corporation
9-11 Midori-Cho 3-Chome Musashinoshi, Tokyo, 180-8585 Japan
{tahara.mitsuho,tateishi.naoki,oimatsu.toshio,
majima.souhei}@lab.ntt.co.jp

Abstract. We propose an improved technique to detect BGP (Border Gateway Protocol) prefix hijacking. BGP prefix hijacking is caused by improper BGP routing information advertisements. When an AS's prefix is hijacked by an improper BGP routing information advertisement from another AS, the hijacked AS can not communicate with other ASs because traffic to the prefix is delivered to the hijacking AS, not the hijacked AS. The method used in major detection systems, such as ENCORE and Keiro-Bugyo, validates each BGP routing information update against BGP routing information stored in IRR (Internet Routing Registry) databases. In order for this method to work, correct BGP routing information must be stored in the IRR. If BGP routing information is advertised to the Internet before being registered with the IRR, it is considered as an instance of prefix hijacking, even if the advertised routing information is valid. We propose a method of detecting prefix hijacking that uses ping tests from two or more ASs. The key idea is that, when an AS receives a BGP update, it executes ping tests to the received prefix; another AS, one that has not received that update also performs similar ping tests. Test results are compared to decide whether prefix hijacking has occurred or not.

Keywords: Border Gateway Protocol (BGP), Prefix Hijacking, Ping Test.

1 Introduction

The Internet is a collection of more than 20,000 autonomous systems (ASs) consisting of a set of routers and networks under the same admission. ASs are connected by dedicated lines or Internet exchanges, and use BGP (Border Gateway Protocol) [1] to exchange their routing information. The routers determine the paths on which to forward IP packets according to the routing information exchanged via BGP.

The routers cannot verify the contents of the BGP messages, and so believe what they hear. This vulnerability allows a malicious AS to advertise wrong routing information that results in IP packets being forwarded along the wrong paths. Prefix hijacking is representative of the communication failure caused by erroneous routing information. When an AS's prefix has been hijacked by an improper BGP routing information advertisement from another AS, the hijacked AS can no longer communicate with other ASs because the traffic to the prefix is delivered to the hijacking AS, not the hijacked AS.

To attack improper routing information, some improved BGP protocols, such as secureBGP [2], soBGP [3], have been proposed. But they have not been deployed in practice, because the digital signature techniques used in these improved BGP protocols require full or at least large-scale deployment to be effective. These improved BGP protocols are future solutions. Therefore, we need detection and recovery methods that do not require modification of the BGP protocol.

The method used in major detection systems, such as ENCORE [4] and Keiro-Bugyo [5], validates each BGP routing information update against BGP routing information stored in IRR (Internet Routing Registry) databases [6,7]. In order for this method to work, correct BGP routing information must be stored in the IRR. If BGP routing information is advertised to the Internet before being registered with the IRR, it is considered as an instance of prefix hijacking, even if the advertised routing information is valid.

In this paper we propose a method of detecting prefix hijacking that uses ping tests from two or more ASs. The key idea is that, when an AS receives a BGP update, it executes a ping test to the received prefix; another AS, one that has not received that update also performs a similar ping test. Test results are compared to decide whether prefix hijacking has occurred or not.

2 Prefix Hijacking and Punching-Hole

An example of prefix hijacking is shown in Fig.1. In this figure, AS65010, AS65020, AS65030, AS65040 and AS65050 exchange routing information with each AS by BGP, yielding the routing tables in each AS's router. The routers forward IP packets by referring to their own routing table. When AS650040 advertises 10.0.0.0/24, which is a part of 10.0.0.0/16 that AS65050 owns, the routing information of 10.0.0.0/24's AS_PATH is (65020 65030 65040) is written in AS65010's routing table. Therefore, traffic to 10.0.0.0/24 is delivered to AS65040 via AS65020, not to AS65050.

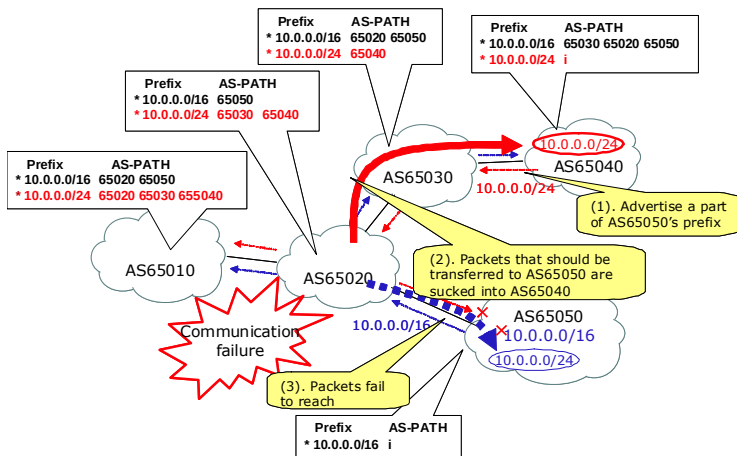


Fig. 1. An example of prefix hijacking

One well-known network technique, punching-hole, is used to achieve network redundancy. A common example of punching hole is explained in Fig.2. Assume that the customer has been assigned 10.0.0.0/24, a part of 10.0.0.0/16 that AS65050 owns, by AS65050. For network redundancy, the customer connects to AS65040, and AS65040 advertises its /24 to other ASes for the customer. If there is a link failure between AS65030 and AS65040, the path from AS6510 to the customer is changed from a path via AS65040 to a path via AS65050. Thus, the customer achieves network redundancy.

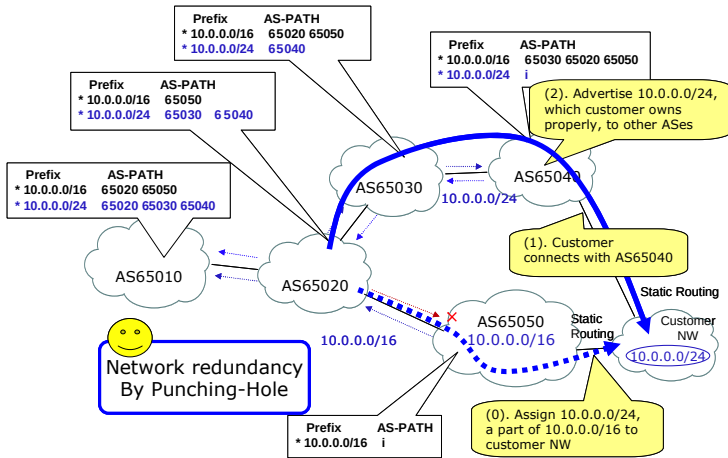


Fig. 2. An example of punching-hole

When prefix hijacking is compared to punching-hole, we find that there is no difference in BGP routing information. Therefore, it is necessary to distinguish prefix hijacking from punching-hole by using information other than BGP routing information.

3 Technique Based on IRR Database

3.1 Detection Algorithm

An example of Route Object stored in IRR [6] is shown in Fig. 3. The "route:" attribute is the address prefix of the route and the "origin:" attribute is the AS number of the AS that originated the route.

The detection algorithm is as follows. First, the route objects are retrieved from IRR by searching with the key of the prefix of received routing information. If the origin AS of received routing information is included in the origin attributes of retrieved route objects, the received routing information is considered to indicate punching-hole. If not, it is considered as prefix hijacking.

```

route: 192.168.0.0/21
descr: Japan Internet Routing Registry Inc. CIDR-block NO.1
origin: AS00000
notify: admin@example.jp
mnt-by: MAINT-AS00000
changed: admin@example.jp 20060801
source: JPIRR

```

Fig. 3. Example of Route Object

Therefore, in Fig. 2, when the route object whose route attribute and origin attribute are 10.0.0.0/24 and AS65040, respectively, is registered in the IRR, the advertised routing information would be considered as punching-hole, not prefix hijacking.

This technique is used widely in ENCORE, Keiro-Bugyo, MyASN, and our project [7].

3.2 Detection Accuracy

We evaluated that technique by using our multiagent-based inter-AS diagnostic system [7]. The outline of our system is shown in Fig. 4. This system consists of intelligent agents located in multiple ASs. Each agent transfers prefix-list to other agents, and requests that the others observe those prefixes. Each requested agent connects to the BGP router in each AS by BGP, and monitors received BGP routing information. When BGP routing information that is on the prefix-list set for observation is detected, each agent informs the requesting agents that suspicious routing information was detected. After receiving the information, the requesting agent queries the IRR database. If the corresponding route object is not registered in IRR, the agent alerts the network operator by raising an alarm of prefix hijacking.

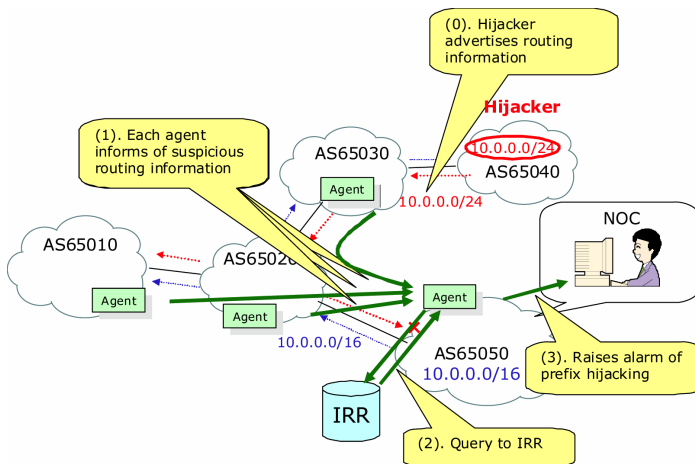


Fig. 4. Outline of multiagent-based inter-AS diagnostic system

By using this system, which consists of 8 agents located in 8 ASs, about 1,200 prefixes were observed for four months.

The alarms of prefix hijacking issued were analyzed through the cooperation of the operators of the 8 ASs. The analyzed result is shown Table 1. 76 alarms were raised in four months. Only 3 alarms were real prefix hijacking events, and the other 73 alarms were punching-hole events. 4 of these 73 were caused by the leakage of route object registration to the IRR, and the rest were caused by mis-configuration of BGP routing.

Table 1. Detection accuracy of the technique based on IRR

Prefix Hijacking		3
Punching-Hole		73
Leakage of route object registration to the IRR	4	
Mis-configuration of BGP routing	69	
Total alarms		76

Therefore, even if the accuracy of the registered information is improved, the punching-hole events caused by mis-configuration of BGP routes will be mis-judged as prefix hijacking. We conclude that the detection accuracy of the technique based on IRR information is low and can not be improved.

4 Technique Based on Ping Tests

4.1 Basic Idea

Our solution is a method that uses ping tests to detect prefix hijacking accurately.

The key idea is that, when an AS receives suspicious BGP routing information that is on the prefix-list to be observed, it executes ping tests to the received prefix; another AS, one that has not received that routing information also performs similar ping tests. The test results of both ASs are compared to decide whether prefix hijacking has occurred or not. If the test results are very similar, it is considered that the ping packets from received and non-received ASs were transferred to the same network. Therefore, the event is treated as punching-hole. If the similarity is low, it is considered that the ping packets from received AS were transferred to a network different from the network to which the ping packets from the non-received AS were transferred. This is treated as indicating prefix hijacking.

The proposed technique is explained in detail in Fig. 5 and 6. The network structures of Fig.5 and Fig.6 are those in Fig. 1 and Fig. 2, respectively. When agent receives suspicious BGP routing information that is on the prefix-list to be observed, each agent executes ping tests to the received prefix, simultaneously. Concretely, if the suspicious prefix is 10.0.0.0/24, ping tests are executed to the 256 addresses from 10.0.0.0 to 10.0.0.255. Of course, it's possible to reduce the target addresses to be tested and the agents to participate in the tests after consideration of the target network load.

In the case of prefix hijacking as shown in Fig.5, when the agents of AS65010, AS6520, and AS5030, which receive suspicious BGP routing information, execute ping tests to the suspicious prefix, ping packets are transferred to AS65040. On the other hand, when the agent of AS65050, which has not received the suspicious BGP routing information, executes ping tests to the same prefix, ping packets are transferred to the network properly, i.e. inside AS65050. Therefore, in the case of prefix hijacking, ping packets from the agents are transferred to different networks, and the results of the ping tests are different. Therefore, if the similarity of ping tests is low, it follows that the event is prefix hijacking.

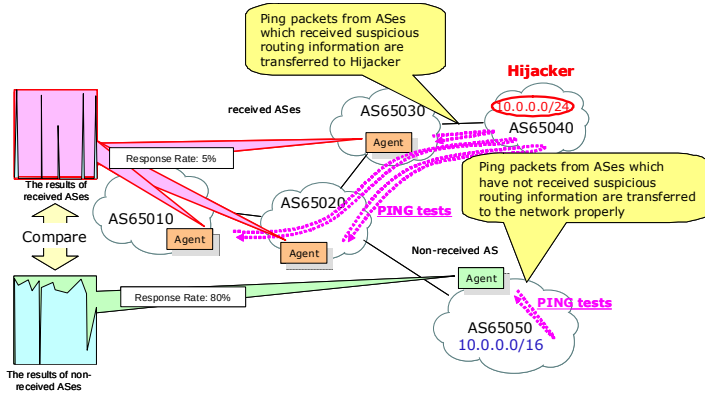


Fig. 5. Ping Tests: Prefix Hijacking Event

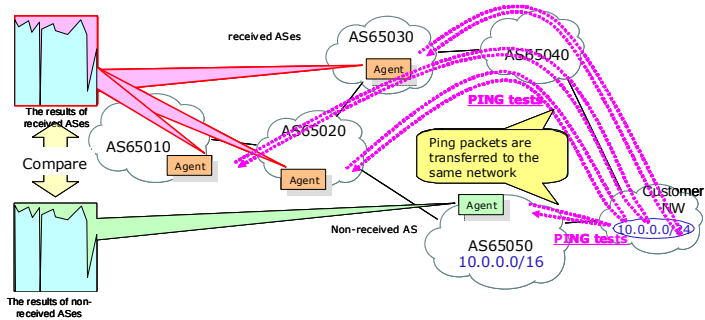


Fig. 6. Ping Tests: Punching-Hole Event

In the case of punching-hole, shown in Fig.6, when the agents of AS65010, AS6520, and AS5030, which receive suspicious BGP routing information, execute ping tests to the suspicious prefix, ping packets are transferred via AS65040 to the customer network. On the other hand, when the agent of AS65050, which did not receive the suspicious BGP routing information, executes ping tests to the same prefix, ping packets are transferred to the same customer network. Therefore, in case of punching-hole, ping packets from all agents are transferred to the same networks, and the results of the ping tests are the same.

By the way, why did not AS65050 receive the suspicious BGP routing information? The reason for this is that an AS configures a route filter to reject routing information about self-AS's prefixes advertised from other ASs generally. Therefore, one or more AS which does not receive the suspicious BGP routing information exists.

4.2 Comparing Ping Test Results

If we assume that the success or failure of a ping test to one address is represented as 1 or -1, respectively, the result of ping tests to N addresses is represented as a vector of N dimensions as follows.

$$X = (1, -1, -1, 1, -1, -1, -1, \dots, -1)$$

The similarity of the results of ping tests from received and non-received agents is given by the numerical similarity of two vectors of X and Y . We use angle as the metric to express the similarity of two vectors numerically. Angle is determined as follows.

$$\text{rad}(X, Y) = \cos^{-1} \frac{\sum x_i y_i}{\sqrt{\sum x_i^2} \sqrt{\sum y_i^2}}$$

4.3 Summary of Proposal

Our proposal is summarized as follows. When agents receive suspicious BGP routing information, i.e. on the prefix-list to be observed, each agent executes ping tests to the received prefix, simultaneously. The angle of two vectors, which are the results of ping tests from received agents and non-received agents, is calculated to decide whether prefix hijacking has occurred or not.

5 Preliminary Experiment

We conducted a preliminary experiment to evaluate the effectiveness of our proposal. The experimental environment is shown Fig 7. As the proposed method, ping tests to the same prefix were executed by two ASs. The target prefixes were 3000 prefixes with different origin ASs extracted from the routing information of AS-X and AS-Y.

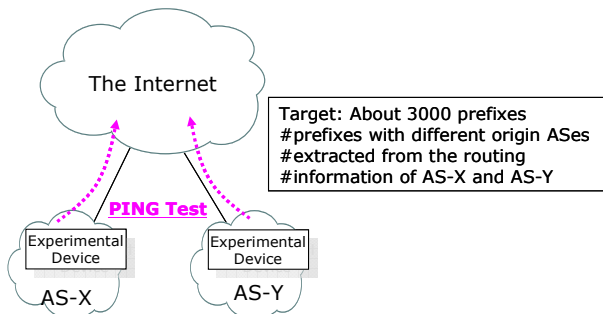


Fig. 7. Environment of preliminary experiment

The histogram of the angle values is shown in Fig 8. The angle was 0 degrees for about one third of the 3000 prefixes, and the majority are within 45 degrees. The maximum value was 129 degrees.

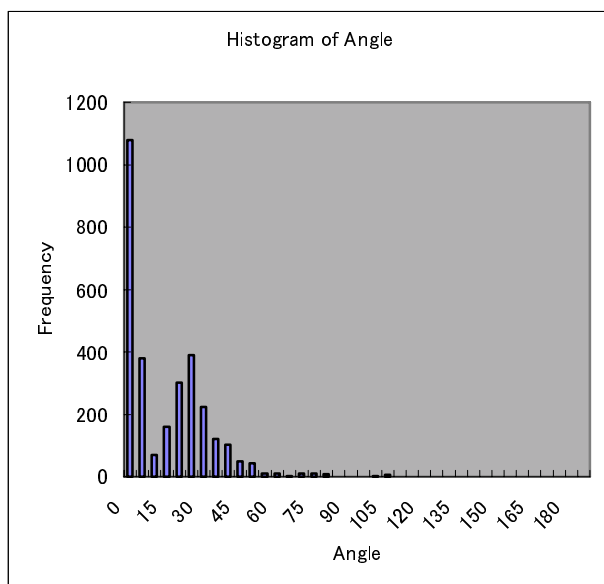


Fig. 8. Histogram of angle of ping tests

From the experiment, samples of 0, 30, 45, 60, 90 and 120 degrees are shown in Table 2.

The number of addresses under prefix-a was 128. The number of addresses in which ping test succeeds at both AS-X and AS-Y is 0. The number of addresses in which ping test fails at both AS-X and AS-Y is 65. The number of addresses in, which ping test succeeds in AS-X and fails in AS-Y is 63. The number of addresses in, which ping test fails in AS-X and succeeds in AS-Y is 0. Therefore, the ping packets from AS-X and AS-Y were obviously transferred to different networks. The angle of 129 degrees shows that the possibility of prefix hijacking is very high.

On the other hand, in the case of prefix-f, the results of ping tests from AS-X and AS-Y correspond completely. Therefore, the ping packets from AS-X and AS-Y are obviously transferred to the same networks. The angle of 0 degrees shows that the possibility of prefix hijacking is very low.

In case of prefix-e, there is only one address that the results of ping tests does not correspond. Its cause is considered as error of ping test. The angle of 30 degrees shows that the possibility of prefix hijacking is low relatively.

In case of prefix-d, the number of addresses in which ping test succeeds at both AS-X and AS-Y is 629, but the number of address in which ping tests correspond is 549. Therefore, it is not possible to decide whether the ping packets from AS-X and

AS-Y are transferred to the same networks or not. The angle of 45 degrees shows that the possibility of prefix hijacking is middle.

Therefore, we conclude that our proposal method have some ability to differentiate prefix hijacking from punching-hole.

Table 2. Excerpt of ping tests

prefix	ip address	both	nothing	x-only	y-only	angle
prefix-a	128	0	65	63	0	129.2894069
prefix-b	256	0	232	24	0	90.70103344
prefix-c	1024	0	1006	18	0	60.87230879
prefix-d	16384	629	15206	38	511	45.28481241
prefix-e	64	3	60	0	1	30.88254849
prefix-f	2048	13	2035	0	0	0

6 Conclusion

This paper proposed a method to accurately detect prefix hijacking by using ping tests, and confirmed its effectiveness by a preliminary experiment. In the near future, we will implement this proposal in a prefix hijacking detection system and evaluate its effectiveness in cooperation with network operators.

Acknowledgement

This paper is the output of a research project for the Ministry of Internal Affairs and Communications (Japan), “Research and Development Relating to the Detection, Recovery, and Prevention of Prefix Hijacking.”

References

1. Rekhter, Li.: A Border Gateway Protocol 4 (BGP-4), RFC (1771)
2. Lynn, C., Mikkelsen, J., Seo, K.: Secure BGP (S-BGP), draft-clynn-s-bgp-protocol-01.txt
3. White, R.: Architecture and Deployment Considerations for Secure Origin BGP (soBGP), draft-white-sobgparchitecture-00.txt
4. Toshimitsu, O., Mitsuho, T., Kazuo, K.: A study on the method to observe hijack route. Technical report of IEICE. TM, TM2004-37
5. <http://www.janog.gr.jp/meeting/janog19/2006/11/jpirr.html>
6. <http://www.radb.net/>
7. <http://www.nic.ad.jp/ja/ip/irr/index.html>
8. <http://www.nic.ad.jp/doc/jpnic-01048.html#3>
9. http://www.soumu.go.jp/menu_02/ictseisaku/ictR-D/jigyuu_ichiran_h18_2.html

Application of Data Mining to Network Intrusion Detection: Classifier Selection Model

Huy Anh Nguyen and Deokjai Choi

Chonnam National University, Computer Science Department,
300 Yongbong-dong, Buk-ku,
Gwangju 500-757, Korea
anhhuy@gmail.com, dchoi@chonnam.ac.kr

Abstract. As network attacks have increased in number and severity over the past few years, intrusion detection system (IDS) is increasingly becoming a critical component to secure the network. Due to large volumes of security audit data as well as complex and dynamic properties of intrusion behaviors, optimizing performance of IDS becomes an important open problem that is receiving more and more attention from the research community. The uncertainty to explore if certain algorithms perform better for certain attack classes constitutes the motivation for the reported herein. In this paper, we evaluate performance of a comprehensive set of classifier algorithms using KDD99 dataset. Based on evaluation results, best algorithms for each attack category is chosen and two classifier algorithm selection models are proposed. The simulation result comparison indicates that noticeable performance improvement and real-time intrusion detection can be achieved as we apply the proposed models to detect different kinds of network attacks.

Keywords: Data mining, Machine learning, Classifier, Network security, Intrusion detection, Algorithm selection, KDD dataset.

1 Introduction

In the era of information society, computer networks and their related applications are becoming more and more popular, so does the potential threat to the global information infrastructure to increase. To defend against various cyber attacks and computer viruses, lots of computer security techniques have been intensively studied in the last decade, namely cryptography, firewalls, anomaly and intrusion detection ... Among them, network intrusion detection (NID) has been considered to be one of the most promising methods for defending complex and dynamic intrusion behaviors.

Intrusion detection techniques using data mining have attracted more and more interests in recent years. As an important application area of data mining, they aim to meliorate the great burden of analyzing huge volumes of audit data and realizing performance optimization of detection rules. Different researchers propose different algorithms in different categories, from Bayesian approaches [8] to decision trees [9, 10], from rule based models [3] to functions studying [11]. The detection efficiencies therefore are becoming better and better than ever before.

However, to the best of our knowledge, never before has it existed a considerable comparison among these classification methods to pick out the best ones that suite the job of intrusion detection. A literature survey that was done by us also indicates a fact that, for intrusion detection, most researchers employed a *single algorithm* to detect *multiple attack categories* with dismal performance in some cases. Report results suggest that much detection performance improvement is possible. In light of the widely-held belief that attack execution dynamics and signatures show substantial variation from one attack category to another, identifying attack category specific algorithms offers a promising research direction for improving intrusion detection performance.

In this paper, a comprehensive set of classifier algorithms will be evaluated on the KDD dataset [2]. We will try to detect attacks on the four attack categories: Probe (information gathering), DoS (deny of service), U2R (user to root), R2L (remote to local). These four attacks have distinct unique execution dynamics and signatures, which motivates us to explore if in fact certain, but not all, detection algorithms are likely to demonstrate superior performance for a given attack category. And from the performance comparison result of the classifiers, we will hereby propose the model for classifier algorithm selection regarding the best performing algorithms for each attack category.

The remainder of this paper is organized as follows. We make a quick and up-to-date literature survey on attempts for designing intrusion detection systems using the KDD dataset in Section 2. Section 3 will detail about our simulation study (classifiers, evaluation setup and performance comparison). Two models for algorithm selection will be proposed in Section 4. Section 5 is the performance comparisons to prove the effectiveness of our models; implementing issues will also be discussed here. Finally, Section 6 will conclude our study and discuss the future works.

2 Related Works on KDD99 Dataset

Agarwal and Joshi [4] proposed a two-stage general-to-specific framework for learning a rule-based model (PNrule) to learn classifier models on a data set that has widely different class distributions in the training data. The PNrule technique was evaluated on the KDD testing dataset which contained many new R2L attacks that were not presented in the KDD dataset. The proposed model was considered well-performed with the TP (*true positive*) rate at 96.9% for DoS, 73.2% for Probe, 6.6% for U2R and 10.7% for R2L attacks. FP (*false positive*) rate was generated at a level of less than 10% for all attack categories except for U2R: an unacceptably high level of 89.5% FP was reported.

Yeung and Chow [5] proposed a novelty detection approach using no-parametric density estimation based on Parzen-window estimators with Gaussian kernels to build an intrusion detection system using normal data only. This novelty detection approach was employed to detect attack categories in the KDD dataset. The technique has surprisingly good reported results: 96.71% of DoS, 99.17% of Probe, 93.57% of U2R and 31.17% of R2L respectively. However, due to the fact that no FP was reported by the authors and a nearly impossible detection rate [13] of 93.57% of U2R category, we really have to question the authentic of the reported numbers.

In 2006, Xin Xu et al. [6] presented a framework for adaptive intrusion detection based on machine learning. Multi-class Support Vector Machines (SVMs) is applied to classifier construction in IDSs and the performance of SVMs is evaluated on the KDD99 dataset. Promising results were given: 76.7%, 81.2%, 21.4% and 11.2% detection rate for DoS, Probe, U2R, and R2L respectively while FP is maintained at the relatively low level of average 0.6% for the four categories. However, this study can only use a very small set of data (10,000 randomly sampled records) comparing to the huge original dataset (5 million audit records). Therefore it is difficult to convince a strict audience about the effectiveness of the method.

Yang Li and Li Guo [7] though realize the deficiencies of KDD dataset, developed a supervised network intrusion detection method based on Transductive Confidence Machines for K-Nearest Neighbors (TCM-KNN) machine learning algorithm and active learning based training data selection method. The new method is evaluated on a subset of KDD dataset by random sampling 49,402 audit records for the training phase and 12,350 records for the testing phase. An average TP of 99.6% and FP of 0.1% was reported but no further information about the exact detection rate of each attack categories was presented by the authors.

Literature survey shows that, for all practical purposes, most researchers applied a single algorithm to address all four major attack categories with dismal performance in many cases. This leads to our belief that different algorithms will perform differently on different attack categories, and this also is the motivation of our study.

3 Empirical Study

In order to verify the effectiveness of different classifiers algorithms for the field of intrusion detection, we will use the KDD99 dataset to make relevant experiments step-by-step. Firstly, we build the experiment evaluation environment with major steps: environment setup, data preprocessing, choosing the data mining software. Secondly, we select a comprehensive set of most popular classifier algorithms, ten distinct widely used classifier algorithms were selected so that they represent a wide variety of fields: Bayesian approaches, decision trees, rule based models and function studying and lazy functions. An overview of how specific values of these algorithms were identified as well as their detection performance will be given. Finally, we come up with the performance comparison between the ten selected classifiers.

3.1 Evaluation Setup

All experiments were performed in a one-year-old computer with the configurations Intel(R) Core(TM)2 CPU 2.13GHz, 2 GB RAM, and the operation system platform is Microsoft Windows XP Professional (SP2). We have used an open source machine learning package – Weka (the latest Windows version: Weka 3.5.7). Weka is a collection of machine learning algorithms for data mining tasks that contains tools for data preprocessing, classification, regression, clustering, association rules, and visualization. This empirical study, however, only deals with a subset of classifier algorithms.

All the machine learning techniques that will be used in this paper are implemented in Weka so that they will be easily and fairly compared to each other.

The dataset to be used in our experiments is the KDD99 labeled dataset. The main reason we use this dataset is that we need relevant data that can easily be shared with other researchers, allowing all kinds of techniques to be easily compared in the same baseline. The common practice in intrusion detection to claim good performance with “live data” makes it difficult to verify and improve previous research results, as the traffic is never quantified or released for privacy concerns. The KDD99 dataset might have been criticized for its potential problems [13], but the fact is that it is the most widespread dataset that is used by many researchers and it is among the few comprehensive datasets that can be shared in intrusion detection nowadays.

As our test dataset, the KDD99 dataset contains one type of normal data and 24 different types of attacks that are broadly categorized in four groups of DoS, Probes, U2R and R2L. The packet information in the original TCP dump files were summarized into connections. This process is completed using the Bro IDS, resulting in 41 features for each connection (and one final feature for classifying, of course). Therefore, each instance of data consists of 41 features and each instance of them can be directly mapped into the point discussed in classifiers algorithms.

Due to the huge number of audit data records in the original KDD99 dataset as shown in Table 1, we have to manually random sample twice from the original KDD99 dataset. For the first time, we extracted 49,596 instances as training set for our experiments. They include 9,841 normal instances, 39,092 DoS instances, 437 Probe instances, 13 U2R instances and 213 R2L instances. Secondly, we extracted 15,437 instances as the independent testing set. By using these two datasets, we thus can effectively evaluate the performance of different methods.

Table 1. Distribution of connection types in KDD99 10% training dataset

Class	Number of records	% of occurrence
Normal	97,277	19.69
DoS	391,458	79.24
Probe	4,107	0.83
U2R	52	0.01
R2L	1,126	0.23
Total	494,020	100.00

Table 2. Distribution of connection types in our actual training data for classifiers evaluation

Class	Number of records	% of occurrence
Normal	9,841	19.84
DoS	39,092	78.82
Probe	437	0.88
U2R	13	0.03
R2L	213	0.43
Total	49,596	100.00

3.2 Classifier Algorithms

3.2.1 BayesNet

BayesNet [8] learns Bayesian networks under the presumptions: nominal attributes (numeric one are predescretized) and no missing values (any such values are replaced globally). There are two different parts for estimating the conditional probability tables of the network. We run *BayesNet* with the SimpleEstimator and K2 search algorithm without using ADTree.

3.2.2 NaïveBayes

The *NaïveBayes* [8] classifier provides a simple approach, with clear semantics, to representing and learning probabilistic knowledge. It is termed *naïve* because it relies on two important simplifying assumptions that the predictive attributes are conditionally independent given the class, and it posits that no hidden or latent attributes influence the prediction process.

3.2.3 J48 (C4.5 Decision Tree Revision 8)

Perhaps *C4.5* algorithm which was developed by Quinlan [9] is the most popular tree classifier. Weka classifier package has its own version of *C4.5* known as *J48*. *J48* is an optimized implementation of *C4.5* rev. 8. *J48* is experimented in this study with the parameters: confidenceFactor = 0.25; numFolds = 3; seed = 1; unpruned = False.

3.2.4 NBTree

NBTree [10] is a hybrid between decision trees and *NaïveBayes*. It creates trees whose leaves are *NaïveBayes* classifiers for the instances that reach the leaf. It is quite reasonable to expect that *NBTree* can outperform *NaïveBayes*; but instead, we may have to sacrifice some speed.

3.2.5 Decision Table

Decision Table [3] builds a decision table majority classifier. It evaluates feature subsets using best-first search and can use cross-validation for evaluation. There is a set of methods that can be used in the search phase (E.g.: BestFirst, RankSearch, GeneticSearch ...) and we may also use Lbk to assist the result. In this experiment, we choose the crossVal = 1; searchMethod = BestFirst and useIBk = False

3.2.6 JRip (RIPPER)

RIPPER [3] is one of the basic and most popular algorithms. Classes are examined in increasing size and an initial set of rules for the class is generated using incremental reduced-error pruning. We evaluate *RIPPER* through *JRip*, an implementation of *RIPPER* in Weka with the parameters: folds = 3; minNo = 2; optimizations = 2; seed = 1; usePruning = true.

3.2.7 OneR

OneR [3] is another basic algorithm using Rule based model. It generates a one-level decision tree expressed in the form of a set of rules that all test one particular

attribute. *OneR* is a simple, cheap method that often comes up with quite good rules for characterizing the structure in data.

3.2.8 Multilayer Perceptron (MLP)

Multilayer perceptron (MLP) [11] is one of the most commonly used neural network classification algorithms. The architecture used for the *MLP* during simulations with KDD dataset consisted of a three layer feed-forward neural network: one input, one hidden, and one output layer. Selected parameters for the model are: learningRate = 0.3; momentum = 0.2; randomSeed = 0; validationThreshold = 20.

3.2.9 SMO

SMO [3] implements the sequential minimal optimization algorithm for training a support vector classifier, using polynomial or Gaussian kernels. *SMO* is evaluated with the following parameters: c = 1.0; epsilon = 1.0E-12; kernel = PolyKernel; numFolds = -1; randomSeed = 1.

3.2.10 LBk

LBk [12] is a lazy classifier algorithm that makes use of the k-nearest-neighbor classifier. In this study, we choose the parameters for *LBk* as follow: k = 1; crossValidate = False; searchAlgorithm = LinearNNSearch; windowSize = 0.

3.3 Performance Comparison

Best performing instances of all the ten classifiers selected in Section 3.2 were evaluated on the KDD dataset. Simulation results are given in the Table 4. To compare the classifiers, we record TP and FP of each algorithm. These parameters will be the most important criteria for the classifier to be consider the best algorithm for the given attack category. Besides, it is also at equal importance to record Average Accuracy (AA = Total correctly classified instances/Total instances) and Training Time (TT) of each algorithm. In the selection process, one algorithm will be disqualified if its AA is too low, despite its outstanding performance in one specific attack category. TT on the other hand, will give us the idea about which algorithm can be implemented in a real-time network intrusion detection system.

Just as we expected, Table 1 shows that no single algorithm could detect all attack categories with a high probability of detection and a low false alarm rate. It strengthen our belief that different algorithms should be used to deal with different types of network attacks. Results also show that for a given attack category, certain algorithms demonstrate superior detection performance compared to others. For DoS category, most algorithms yield very high TP rates – averagely 95%. *NaïveBayes* is the only one that lags behind as it gives a TP at 79.2%. But for Probe attacks, *NaïveBayes* outperforms the others with its FP at 94.8%; *BayesNet* and *Decision Table* both have impressive performance for this category at 83.8%. In U2R attacks, *BayesNet* and *Decision Table* are the best two classifiers with FP at 30.3% and 32.8% respectively. And for the case of R2L attacks, only *OneR* could produce about 10% of attacks while the others just lag behind with inappreciable results.

Table 4. Performance comparison of the ten classifier algorithms – TP, FP and AA is measured in %. TT is measured in second.

Classifier Category	Classifier Algorithm		DoS	Probe	U2R	R2L
Bayes	BayesNet	TP	94.6	83.8	30.3	5.2
		FP	0.2	0.13	0.3	0.6
		AA	90.62			
		TT	6.28			
	NaïveBayes	TP	79.2	94.8	12.2	0.1
		FP	1.7	13.3	0.9	0.3
		AA	78.32			
		TT	5.57			
Trees	J48	TP	96.8	75.2	12.2	0.1
		FP	1	0.2	0.1	0.5
		AA	92.06			
		TT	15.85			
	NBTree	TP	97.4	73.3	1.2	0.1
		FP	1.2	1.1	0.1	0.5
		AA	92.28			
		TT	295.88			
Rules	Decision Table	TP	97.0	57.6	32.8	0.3
		FP	10.7	0.4	0.3	0.1
		AA	91.66			
		TT	66.24			
	JRip	TP	97.4	83.8	12.8	0.1
		FP	0.3	0.1	0.1	0.4
		AA	92.30			
		TT	207.47			
	OneR	TP	94.2	12.9	10.7	10.7
		FP	6.8	0.1	2	0.1
		AA	89.31			
		TT	3.75			
Functions	MLP	TP	96.9	74.3	20.1	0.3
		FP	1.4	0.1	0.1	0.5
		AA	92.03			
		TT	350.15			
	SMO	TP	96.4	74.3	13.3	0.1
		FP	0.8	0.3	0.1	0.4
		AA	91.65			
		TT	192.16			
Lazy	LBk	TP	96.7	72.4	22.3	7.8
		FP	0.8	0.2	0.1	0.6
		AA	92.22			
		TT	10.63			

4 Classifier Selection Model

After experiencing the performance comparison with the ten classifier algorithms, we generalize the empirical results with a model for algorithm selection. Observation from Table 4 suggests that for a given attack category, certain subset of classifier

algorithms offer superior performance over the others. It is quite reasonable to expect much performance improvement as we can select the best classifier candidate for a specific attack category at hand. Section 3 identified the best algorithms for each attack categories: *JRip* for DoS and Probe, *Decision Table* for U2R and *OneR* for R2L. We then propose a model for classifier selection as in Fig. 1(a).

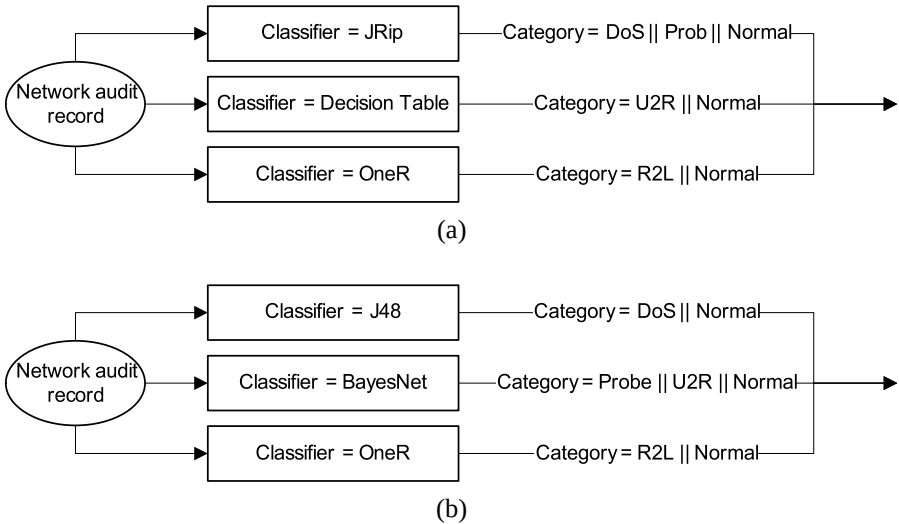


Fig. 1. (a). Parallel model for classifier selection. **(b).** Parallel model for real-time application classifier selection

About the application for the model proposed from the Fig. 1(a), it is expected that network intrusion detection systems with data mining capability should be flexible in choosing the classifying technique that best to deal with the attack at hand. Although detection rate improvement is something we expect, it is at equal importance to judge whether the selected algorithms can be implemented in real-time NIDs. The original need of network intrusion detection actually comes from the commercial world, not from the scientific community. Companies will pay more attention to classifier algorithms that can not only effectively detect network intrusion, but also can detect them in a really short time. Algorithms with high detection rate but end up consuming too much time in fact only are used by researchers in laboratories. Therefore, we suggest another model for real-time algorithm selection as in Fig. 1(b). The best algorithms with low TT for each attack category: *J48* for DoS, *BayesNet* for Probe and U2R and *OneR* for R2L. This model may have significant meaning for companies seeking for implementing real-time IDSs, or researchers that are currently working on developing light-weight data-mining algorithms.

5 Model Evaluation and Discussion

Table 5 shows the performance comparison of the two proposed multi-classifier models with KDD Cup Winner. The results suggest that the two proposed models showed

Table 5. Performance comparison between the two models and KDD Cup Winner

		DoS	Probe	U2R	R2L
KDD Cup Winner	TP	97.10	83.30	13.20	8.40
	FP	0.60	0.30	0.03	0.05
Model 1(a)	TP	97.40	83.80	32.80	10.70
	FP	0.30	0.10	0.30	0.10
Model 1(b)	TP	96.80	83.80	30.30	10.70
	FP	1.00	0.13	0.30	0.10

minor improvement in TP for DoS and Probe; and significant improvement for U2R and R2L attack categories. Also, FP was reasonably small for all attack categories.

Despite the superior in numeric comparison between the proposed models and other approaches, one may have questions on some potential problems when the models are practically deployed in real systems: (1) Deployment of a system with multiple algorithms is inflexible since we may have to hardcode the algorithms. And if we have a better dataset than KDD99, the one used in this paper, then the best classifiers may be different and changing the classifiers will become such annoying. (2) Another problem when the models are implemented is the resource requirements. Consider ingress points of multi-gigabit networks, running multiple algorithms for intrusion detection at these locations may hurt the entire system's network performance. (3) And finally, a comparison between the proposed models and a multiple classifiers selection (MCS) system may be made.

In fact, the above mentioned issues may be well solved, one way or another. (1) To our best knowledge, KDD99 is the only dataset which is well tailored for data mining. There exist many other datasets for network intrusion detection (DRAPA, MAWI, NLANR ...). But they are all "raw" network data, which will require a lot of process before they can be used for data mining applications. Suppose that we have another dataset which are more representative than KDD99 then the best classifiers may indeed be different. But if the system is flexibly designed (no such thing as "hardcode the models"), then modifying the classifier selection models will be easy. (2) Considering the resource problem, x3 or x4 processing time will not exist in reality since the models are configured in a parallel basis, which means that classifiers can be implemented by parallel processors to achieve real-time performance. (3) Finally, MCS systems are still currently designed in the labs. Researchers are proposing different systems. But there exists many problems and currently, none can achieve a stable system with optimistic performance yet. System using the proposed models will surely not be as dynamic as a MCS system, but it can provide stable and reliable performance.

6 Conclusion

For the contribution of this paper, firstly we made an up-to-date survey on recent studies about network intrusion detection that was evaluated with KDD99 dataset. We then use Weka to bring out an extensive performance comparison among the most popular classifier algorithms. Finally, two models for algorithm selection are proposed with great

promise for performance improvement and real-time systems application. However, we are fully aware of problems that have been cited with the KDD99 dataset [13] and strongly discourage its further use in developing network intrusion detection data mining algorithms. The reason we use KDD99 in this paper, as explained, is because we need a baseline to evaluate different algorithms and compare our work with the others. For the future directions, we would like to evaluate our work on another dataset. Besides, we would also like to make real implementations of our algorithm selection models to practically experiment its effectiveness.

Acknowledgement

This research was supported by the Industry Promotion Project for Regional Innovation. The authors would like to thank Prof. Park Hyukro and anonymous reviewers for their valuable comments and suggestions to this paper.

References

1. Weka – Data Mining Machine Learning Software, <http://www.cs.waikato.ac.nz/ml/weka/>
2. KDD Cup 1999 Data, <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>
3. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques, 2nd edn. Morgan Kaufmann, San Francisco (2005)
4. Agarwal, R., Joshi, M.V.: PNrule: A New Framework for Learning Classifier Models in Data Mining. Tech. Report, Dept. of Computer Science, University of Minnesota (2000)
5. Yeung, D.Y., Chow, C.: Prazen-window Network Intrusion Detectors. In: 16th International Conference on Pattern Recognition, Quebec, Canada, pp. 11–15 (August 2002)
6. Xu, X.: Adaptive Intrusion Detection Based on Machine Learning: Feature Extraction, Classifier Construction and Sequential Pattern Prediction. International Journal of Web Services Practices 2(1-2), 49–58 (2006)
7. Li, Y., Guo, L.: An Active Learning Based TCM-KNN Algorithm for Supervised Network Intrusion Detection. In: 26th Computers & Security, pp. 459–467 (October 2007)
8. John, G.H., Langley, P.: Estimating Continuous Distributions in Bayesian Classifiers. In: Proc. of the 11th Conf. on Uncertainty in Artificial Intelligence (1995)
9. Quinlan, J.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo (1993)
10. Kohavi, R.: Scaling up the accuracy of naïve-bayes classifier: A decision-tree hybrid. In: Proc. of the 2nd International Conference on Knowledge Discovery and Data Mining, pp. 202–207. AAAI Press, Menlo Park (1996)
11. Werbos, P.: Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences. PhD Thesis, Harvard University (1974)
12. Aksoy, S.: k-Nearest Neighbor Classifier and Distance Functions. Technical Report, Department of Computer Engineering, Bilkent University (February 2008)
13. Sabhnani, M., Serpen, G.: Why Machine Learning Algorithms Fail in Misuse Detection on KDD Intrusion Detection Dataset. In: Intelligent Data Analysis, vol. 6 (June 2004)

Detection of Leaps/sLumps in Traffic Volume of Internet Backbone

Yutaka Hirokawa¹, Kimihiro Yamamoto¹, Shigeaki Harada²,
and Ryoichi Kawahara²

¹ NTT Information Sharing Platform Laboratories,
9-11, Midori-Cho 3-Chome, Musashino-shi, Tokyo, 180-8585 Japan

² NTT Service Integration Laboratories,
9-11, Midori-Cho 3-Chome, Musashino-shi, Tokyo, 180-8585 Japan
{hirokawa.yutaka,yamamoto.kimihiro,harada.shigeaki,
kawahara.ryoichi}@lab.ntt.co.jp

Abstract. This paper focuses on detecting anomalies in Internet backbone traffic. To monitor traffic on a scale of several terabits per second, we need to divide the time series data of a traffic volume into many slices. Therefore, we need to monitor a lot of traffic data. However, adjusting an appropriate threshold for each traffic time series data individually is difficult. To solve this problem, we propose an anomaly-detection algorithm that does not need parameters to be set for each time series data. This algorithm operates accurately with low computational complexity. A side-by-side test demonstrated that the accuracy of the algorithm was higher than that of the conventional method. Moreover, the necessary learning period of the algorithm was shorter than that of the conventional method.

Keywords: Internet backbone, traffic volume, anomaly detection.

1 Introduction

1.1 Background

This research aims to detect the point at which traffic volume increases or decreases greater than the usual amount in a short time from among the time series data of traffic volume in the Internet backbone. We call a rapid change in traffic volume a *Traffic Leap/sLump*.

Twenty-four-hour traffic trends monitored at JPNAP[6], which is the largest Internet exchange in Japan, are shown in Figure 1. The y-axis of these graphs indicates traffic volume (in bps). A spike type of increase in traffic volume is shown in the left graph. A square-wave type of decrease and increase in traffic volume is shown in the right graph. Changes in traffic volume like these are assumed to be traffic spikes that need to be detected.

Monitoring the quantity of traffic that flows in the network in an ISP (Internet Service Provider)/hosting provider/backbone telecommunications provider is necessary. When a change in traffic volume is generated, in which the quality of the network

service is negatively affected, the network operator needs to detect the change quickly and remove the cause.

Note that we aim at detection of anomalous traffic before analyzing it. By detecting Traffic Leaps/sLumps, network operators can find the symptom of an unknown anomaly that cannot be detected using signature detection and the change in traffic volume caused by a breakdown of equipment or a traffic shift. At the next detailed analysis phase, network operators analyze traffic behavior to determine whether the first detection indicates with certainty an incident affecting actual operation or not. Therefore, suppression of the number of false detections is necessary to reduce the effort of network operators.

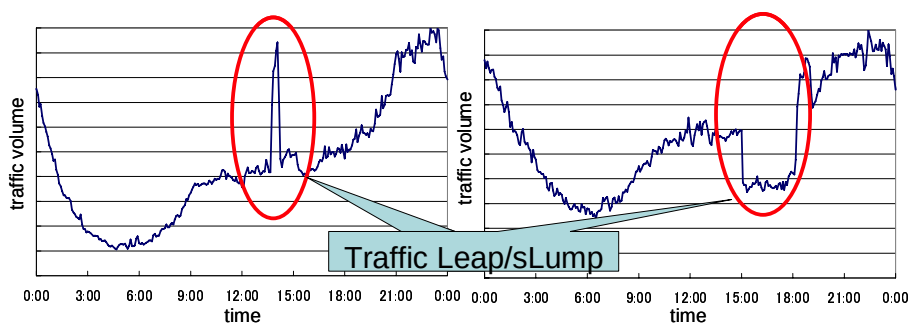


Fig. 1. Traffic Leap and sLump that need to be detected

The amount of traffic on the Internet backbone increases every year. For example, even if there is an increase of several Gbps of traffic volume caused by a DoS attack, detecting that increase in traffic is difficult when the increase is invisible in a regular change of several hundred Gbps of traffic that flows in an ISP backbone. To solve the problem of not being able to detect a Traffic Leap/sLump caused by a DoS attack because the Traffic Leap/sLump is invisible in a regular change of backbone traffic, the backbone traffic is monitored by dividing the traffic into a granularity at which the detection of the Traffic Leap/sLump is possible[4][5]. The simplest method of detecting a Traffic Leap/sLump from the traffic time series data is to set a threshold at a given value and to confirm that a Traffic Leap/sLump occurred when the amount of traffic exceeds the threshold. However, characteristics of the usual traffic volume and fluctuation band are different in each monitored traffic time series data, so the appropriate threshold for each traffic time series data is different. Adjusting an appropriate threshold for each traffic time series data individually is difficult.

1.2 Paper Outline

We propose a learning algorithm that dynamically sets the threshold so that setting parameters of each time series traffic data is not necessary. The remainder of this paper is organized as follows. Section 2 gives features of the Holt-Winters method, which is the most well-known technique, and problems using the method. Section 3 describes details of the TLD algorithm that we propose. We summarize the result of comparing the TLD algorithm and the Holt-Winters method in Section 4 and conclude the paper in Section 5.

2 Problem with Existing Technique

The forecast of the traffic time series data by the Holt-Winters method is a well-known technique to detect anomalies in traffic volume[2][3]. The Holt-Winters method is used with traffic monitoring systems such as RRDtool[7][8]. The Holt-Winters method considers the seasonal trend when forecasting time series data.

When time series data $x_t, x_{t-1}, \dots$ that contain seasonal trends of cycle m are given, the Holt-Winters method calculates the forecast value $\hat{x}_{t+1}$ of x_{t+1} at time $t+1$ by using the following recurrence formula.

$$\hat{x}_{t+1} = a_t + b_t + c_{t+1-m} \quad (1)$$

$$a_t = \alpha(x_t - c_{t-m}) + (1 - \alpha)(a_{t-1} + b_{t-1}) \quad (2)$$

$$b_t = \beta(a_t - a_{t-1}) + (1 - \beta)b_{t-1} \quad (3)$$

$$c_t = \gamma(x_t - a_t) + (1 - \gamma)c_{t-m} \quad (4)$$

The term c_t indicates the change in the seasonal trend of time series data every cycle. The term b_t indicates whether the time series data tends to increase or decrease in a linear trend. The term a_t indicates the baseline of the change in the time series data from which seasonal trends and linear trends are removed.

The threshold to detect a Traffic Leap/sLump is set on the basis of the difference between the forecast value $\hat{x}_t$ and monitored value x_t . First, prediction error d_t is calculated using Equation (5).

$$d_t = \gamma|x_t - \hat{x}_t| + (1 - \gamma)d_{t-m} \quad (5)$$

The value of parameter γ in this expression is equal to γ in Equation (4). The threshold in the time series data of x_t is expressed in Equation (6).

$$(\hat{x}_t - \delta \bullet d_{t-m}, \hat{x}_t + \delta \bullet d_{t-m}) \quad (6)$$

The coefficient δ is used to calculate the width of the threshold. The accuracy of the Traffic Leap/sLump detection improves when δ is set to a small value, and a lot of false detection of points other than those of a Traffic Leap/sLump also occurs. False positive detections decrease when δ is set to a large value, and a lot of small Traffic Leaps/sLumps are also overlooked. Therefore, to calculate the value of δ according to a network management operator policy Traffic Leaps/sLumps need to be detected.

When the Holt-Winters method is used, setting parameters α , β , and γ in Equations (2), (3), and (4), respectively, is necessary. The accuracy of detection greatly depends on the parameter setting. There is a technique mounted in GNU R[9] statistical processing software that dynamically sets α , β , and γ to minimize the sum of the square of the difference between a past forecast value and the monitored value shown in Equation (7). However, using that equation to detect Traffic Leaps/sLumps in a lot of time series data is difficult because the calculation takes an extremely long time and uses a lot of memory.

$$\sum_{t=1}^n (x_t - \hat{x}_t)^2 \quad (7)$$

Therefore, we propose the Traffic Leap/sLump Detection (TLD) algorithm, which does not require setting the parameters of each time series data to resolve the issues described in the above paragraph.

3 TLD Algorithm

3.1 Outline of TLD Algorithm

We propose the TLD algorithm, which does not need to set parameters like the Holt-Winters method and detects Traffic Leaps/sLumps by a simple calculation. Moreover, the TLD algorithm enables high-accuracy detection by a simple calculation.

The flow of the TLD algorithm is shown below.

Flow of TLD algorithm

```

New traffic data  $x_t$  input;
 $y_t = |x_t - x_{t-1}|$  is calculated;
The value of  $y_t$  is stored;
 $z_t = y_t / \bar{y}_t$  is calculated;
The value of  $z_t$  is stored;
if ( $z_t > \text{Threshold}$ ) {
     $x_t$  is regarded as Traffic Leap/sLump;
}
 $\bar{y}_t$  is calculated again;
Threshold is calculated again;
t=t+1;

```

First, when new traffic data x_t is received in t , the TLD algorithm calculates y_t . Moreover, the value of y_t is stored. Next, the TLD algorithm calculates z_t by using y_t and the mean value $\bar{y}_t$ of y_t . The value of z_t is stored. The TLD algorithm sets the threshold of z_t and detects Traffic Leaps/sLumps. If z_t exceeds the threshold, x_t is regarded as a Traffic Leap/sLump.

3.2 Removal of Parameter by Simplification of Forecast

When the parameter of the Holt-Winters method was set to minimize the difference between the forecast value $\hat{x}_t$ and the monitored value x_t , we paid attention to the relationship $\hat{x}_t \approx x_{t-1}$. The Holt-Winters method determines the threshold by using the difference between monitored value x_t and forecast value $\hat{x}_t$ obtained by calculations in which parameters are set and detects Traffic Leaps/sLumps. We propose to use x_{t-1}

instead of forecast value $\hat{x}_t$. The TLD algorithm detects Traffic Leaps/sLumps by applying difference y_t , which is defined in Equation (8), to the traffic time series data.

$$y_t = |x_t - x_{t-1}| \quad (8)$$

As a result, the TLD algorithm detects Traffic Leaps/sLumps without setting the parameter of each time series data.

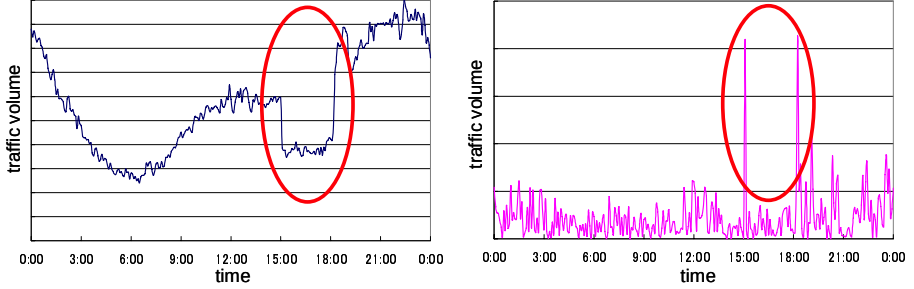


Fig. 2. Difference between two adjacent time points of traffic time series data

In Figure 2, the right graph indicates the difference between two adjacent time points of the traffic time series data of the left graph. The influence of the traffic volume of x_t can be removed by calculating the difference between x_t and x_{t-1} . Two big spikes appear in the time series data shown in the right graph of Figure 2 when the original time series data changes rapidly.

Even though the Holt-Winters method sets the threshold of the time series data, we attempted to improve accuracy by setting the threshold of difference y_t .

3.3 Improvement in Accuracy by Removal of Changes between Day and Night Traffic

The traffic change at night is greater than that of the morning and afternoon, but the difference time series data, y_t , shown in Figure 2 is calculated without considering the seasonal trend. We proposed a method that enables a higher accuracy of detection by removing the influence of the change between day and night traffic and distinguishing the difference between Traffic Leaps/sLumps and usual changes in traffic.

The TLD algorithm suppresses the difference in changes between day and night traffic by dividing y_t by the normal change in that time. The TLD algorithm divides y_t by the mean value of the most recent M , which is a normal change in the traffic during that time.

$$\bar{y}_t = \sum_{k=t-1}^{t-1-M} y_k / M \quad (9)$$

$$z_t = y_t / \bar{y}_t \quad (10)$$

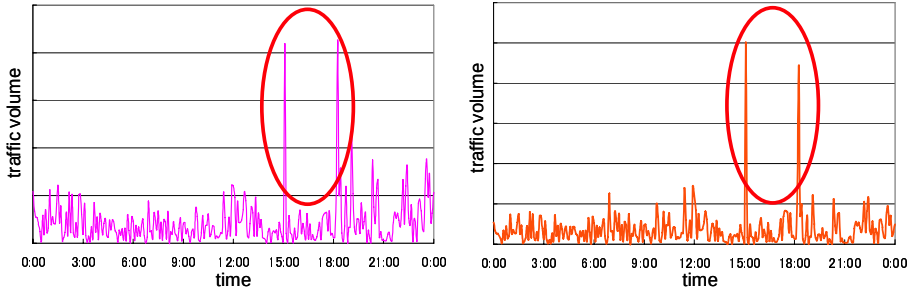


Fig. 3. Correction time series data

We define the new time series data, *correction time series data*, z_t , obtained by dividing the difference y_t by the mean value $\bar{y}_t$, as shown in Equation (10) and the right graph of Figure 3. The TLD algorithm sets the threshold to z_t and detects Traffic Leaps/sLumps.

We verified the best value of M by using actual traffic data aggregated every five minutes while changing the value of M . As a result, the effect of reducing the change between day and night by dividing the mean value was not achieved because previous values strongly influenced mean value $\bar{y}_t$, and $\bar{y}_t$ did not increase even if the change in y_t increased when the value of M was too large.

We set M to two hours ($M = 24$), which is the best value in the TLD algorithm considering the number of false positive (FP) and false negative (FN) detections when experimenting by using actual traffic data.

3.4 Method of Setting Threshold in TLD Algorithm

The TLD algorithm that we propose sets the threshold of correction time series data z_t and detects z_t points that exceed the threshold as Traffic Leaps/sLumps. The difference between Traffic Leaps/sLumps and usual traffic is obvious in z_t . Therefore, setting the threshold of z_t becomes easy. We propose a threshold shown by Equation (11) where the threshold is set to a multiple of the mean of z_t as the best method of setting the threshold for the TLD algorithm. When we verified some methods including the Bollinger Band method[1] by using actual traffic data, the number of FPs in the case of using Equation (11) was less than that of the other methods.

$$\text{Threshold} = \delta \bullet \bar{z}_t \quad (11)$$

4 Comparison between TLD Algorithm and Holt-Winters Method

4.1 Experiment Environment

We compared the accuracy of the TLD algorithm and that of the Holt-Winters method.

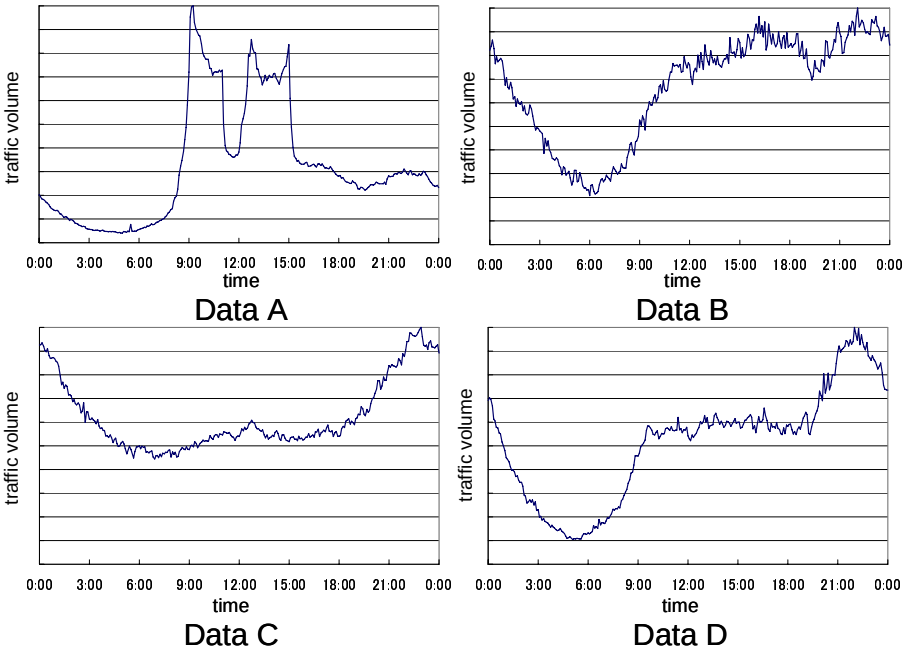


Fig. 4. Traffic data used for experiment

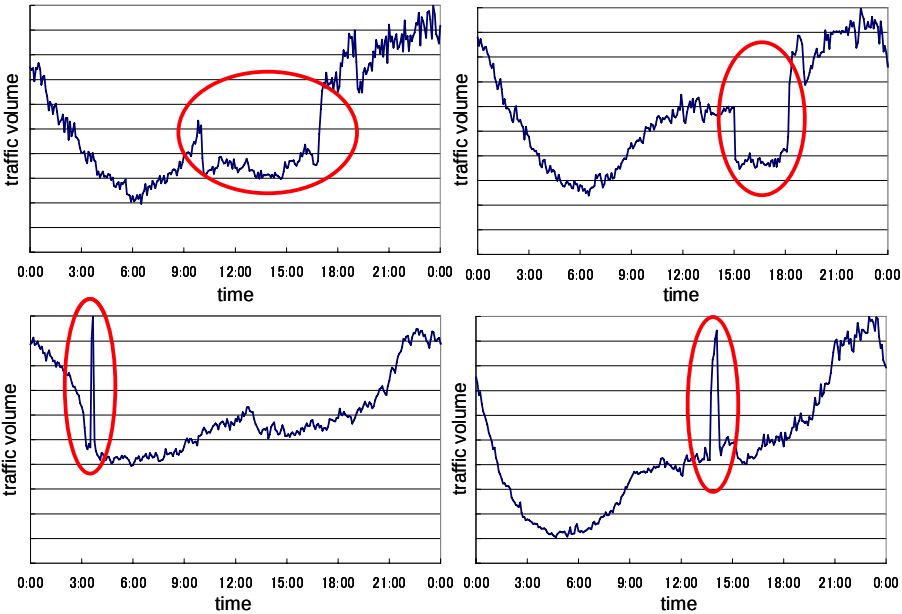


Fig. 5. Examples of Traffic Leaps/sLumps

We used four kinds of traffic data collected for about two months that had been monitored on JPNAP[6] for the comparison. These traffic data were aggregated every five minutes by RRDtool. The graph of features of traffic data collected for a day is shown in Figure 4. Data A, B, C, and D shown in Figure 4 are the traffic data observed at different points. There are a lot of changes in these traffic data that are different from usual, as shown in the examples of Figure 5. The operators in JPNAP selected suspicious points that need to be analyzed in detail. We defined these points as Traffic Leaps/sLumps. Traffic Leaps/sLumps that had not been detected by either algorithm were counted as FNs. Moreover, all Traffic Leap/sLump detections other than these Traffic Leaps/sLumps were assumed to be false detections and counted as FPs.

4.2 Accuracy Comparisons of Detection

We compared the accuracy of the TLD algorithm with that of the Holt-Winters method in detecting Traffic Leaps/sLumps.

The accuracy of the Holt-Winters method changes depending on seasonal cycle m and parameters α , β , and γ in Equations (2), (3), and (4), respectively. Before the experiment, we measured FNs and FPs and obtained the best values for α , β , and γ from actual traffic data. We fixed the values of α , β , and γ to the best values and experimented.

We changed coefficient δ , from 3 to 7 in steps of 1 in which the width of the threshold was calculated. We compared the accuracies of the two methods by using this value in both algorithms because the values of FN and FP were the smallest in the case of $\delta = 6$ in both algorithms.

Table 1. Number of FNs of both algorithms

	Holt-Winters method	TLD algorithm
Data A (Total Traffic Leaps/sLumps: 4)	3	0
Data B (Total Traffic Leaps/sLumps: 14)	2	4
Data C (Total Traffic Leaps/sLumps: 1)	0	0
Data D (Total Traffic Leaps/sLumps: 5)	0	0

Table 2. Number of FPs per day of both algorithms

	Holt-Winters method	TLD algorithm
Data A	0.78	2.14
Data B	0.59	0.17
Data C	0.51	0.07
Data D	0.86	0.64

The comparison results of FN and FP are shown in Tables 1 and 2, respectively.

There is no difference in the frequency of FN of both algorithms. However, each cause is different. In traffic with greatly different features on weekdays and holidays, as shown in Figure 6, FNs of the Holt-Winters method were generated by applying the threshold obtained by studying the traffic on the weekday to the traffic on holidays because the Holt-Winters method considers a seasonal cycle.

The TLD algorithm may detect Traffic Leaps/sLumps by using only the data immediately before traffic data. Therefore, the TLD algorithm does not cause such overlooking of Traffic Leaps/sLumps.

When Traffic Leaps/sLumps that should be detected increase or decrease more gradually than traffic in an interval, the difference in total traffic volume between two time intervals becomes small. Therefore, the TLD algorithm might overlook Traffic Leaps/sLumps because the TLD algorithm detects Traffic Leaps/sLumps based on the difference in the amount of total traffic for each interval.

For data A, there were a very large number of misdetection frequencies of the TLD algorithm. The reason is that Data A is traffic, as shown in Figure 4, where a rapid change appears regularly and periodically. The TLD algorithm does not consider the cycle variation and only studies data immediately before the traffic data, so these periodic sudden change movements are detected as Traffic Leaps/sLumps.

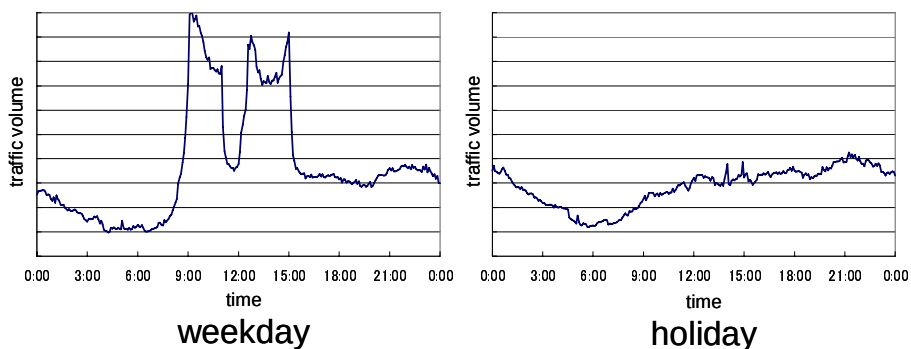


Fig. 6. Traffic of Data A on weekday and holiday

In three data other than data A, the frequency of FPs of the TLD algorithm was less than that of the Holt-Winters method. A long study period, which depends on the cycle of the traffic data, is needed to generate enough accuracy because the Holt-Winters method considers the cycle variation, and when the study period is short, false detection occurs. The TLD algorithm only uses data immediately before traffic data, so Traffic Leaps/sLumps are normally detected using a short data collection period.

5 Conclusion

We proposed the TLD algorithm, which detects Traffic Leaps/sLumps in the traffic volume of time series data neither needing to set parameters for each time series data nor considering the cycle variation. As a result of the comparison with the Holt-Winters method, which is a conventional method, the TLD algorithm detected Traffic Leaps/sLumps where the number of FPs was less than that of the Holt-Winters method excluding traffic that has a periodic sudden change in movement within a time interval.

As future tasks, we need to distinguish the sudden change movement that appears periodically causing false detection of Traffic Leaps/sLumps when using the TLD algorithm and add a mechanism to prevent that. In addition, we plan to evaluate the TLD algorithm when using it for traffic data with different volume and speed from the traffic data used in this experiment. We need to examine the influence on the accuracy of Traffic Leap detection, when the aggregation interval of the traffic data is changed, and the best aggregation interval.

Acknowledgements

This work was supported by the Ministry of Internal Affairs and Communications, Japan.

References

1. Bollinger, J.: Bollinger On Bollinger Bands. McGraw-Hill, New York (2001)
2. Brutlag, J.D.: Aberrant Behavior Detection in Time Series for Network Monitoring. In: Proceedings of the USENIX Fourteenth System Administration Conference LISA XIV, New Orleans, LA (December 2000)
3. Barford, P., Kline, J., Plonka, D., Ron, A.: A Signal Analysis of Network Traffic Anomalies. In: Internet Measurement Workshop 2002 (2002)
4. Ishibashi, K., Kawahara, R., Mori, T., Kondoh, T., Asano, S.: Effect of sampling rate and monitoring granularity on anomaly detectability. In: 10th IEEE Global Internet Symposium (2007)
5. Kawahara, R., Mori, T., Kamiyama, N., Harada, S., Asano, S.: A study on detecting network anomalies using sampled flow statistics. In: IEEE SAINT 2007 Workshop (2007)
6. JPNAP WWW page, <http://www.mfeed.co.jp/jpnap/>
7. Oetiker, T.: The rrdtool manual, <http://oss.oetiker.ch/rrdtool/>
8. Brutlag, J.D.: Notes on RRDTOOL implementation of Aberrant Behavior Detection, http://cricket.sourceforge.net/aberrant/rrd_hw.htm
9. GNU R WWW page, <http://www.r-project.org/>

A Sampling Method for Intrusion Detection System

Zhuo Ning^{1,2} and Jian Gong^{1,2}

¹ School of Computer Science and Engineering, Southeast University, Nanjing 210096, China

² Jiangsu Provincial Key Laboratory of Computer Network Technology, Nanjing 210096, China
{zhning, jgong}@njnet.edu.cn

Abstract. It is well known that Intrusion Detection System (IDS) does not scale well with Gigabit links. Unlike the other solutions that try to increase the performance of IDS by the distributed architecture, we develop a novel sampling method IDSampling whose sampling rate is adaptive to the memory bottleneck consumption to capture attack packets as many as possible by analyzing characteristics of the attack traffic. IDSampling applies a single sampling strategy based on four traffic feature entropies when large-scale traffic anomaly occurs, and another complicated one instructed by the feedback of the following detection results by default. The results of experiment show that IDSampling can help IDS to remain effective even when it is overloaded. And compared with the other two notable sampling method, packet sampling and random flow sampling, IDSampling outperforms them greatly, especially in low sampling rate.

Keywords: Intrusion detection system, sampling, multistage bloom filter, feature entropy.

1 Introduction

The Misuse Intrusion Detection System(MIDS) is prevailing in practice for it can provide explicit alerts to users in low false rate. However, it is computationally infeasible to deal with gigantic operations of data storage and analyze in Gigabit link speed. To address this problem, many works have been investigated.

Early work focuses on how to augment IDS processing power. [1~3] provide hardware solutions that enhance the performance of IDS by parallel computing, but hardware solutions are hard to be popular due to their high cost. [4,5] propose distributed IDS architectures that distribute traffic to a bunch of IDS detectors using payload balance policy and integrate several IDSes as a whole. However, the distributed IDS architecture also has several cons. Firstly, the distributing policy for IDS is required not only to distribute the traffic as even as possible but also to assign the same session to the same IDS, otherwise the context of the attack will be damaged and IDS won't detect it well. So at some special scenarios, such as DDOS with the same source IP and destination IP, the single IDS detector also has the performance problem as before. It indicates that the distributed IDS architecture can not solve the performance-accuracy imbalance problem radically. Secondly, it is rather hard to configure how many IDSes are reasonable to be packed as a whole because either it is

suboptimal in low traffic volume or it has the same resource consumption difficulties in high traffic volume. Finally, the maintenance and updating of the distributed architecture are undoubtedly more complicated than that of an IDS.

Compared with the aforementioned work, the following addresses performance limits by sampling, but all focuses on anomaly detection^[6-11]. In [6], A.Lakhina proposes a notable anomaly detection method using multiway subspace method to analyze netflow data. In [7], Mai investigates how packet sampling impacts three specific portscan detection methods, TRWSYS, TAPS and entropy-based profiling method. Recently, the work is extended to analyze the impact of other sampling schemes in [8]. It demonstrates further that the random flow sampling is better than packet sampling in anomaly detection. And the findings of [9] suggest that entropy summarizations are more resilient to sampling than volume metrics. However, it is still an open problem whether sampling solutions are sufficient in network-wide intrusion detection.

Unlike the aforementioned sampling methods which apply the traditional packet sampling or flow sampling directly, we propose a novel sampling method IDSampling which is adaptive to the consumption of the memory bottleneck to cover not only anomaly detection but also misuse detection. IDSampling profiles the traffic and applies the single sampling strategy based on four traffic feature entropies when large-scale traffic anomaly occurs, and adopts another complicated one instructed by the feedback of the following detection results by default. The aim is to capture attack packets as many as possible under restricted sampling rate. The results of experiment show that IDSampling can help overloaded IDS remain effective and it outperforms traditional sampling methods greatly.

The rest of this paper is organized as follows. We analyze the characteristics of the attack flow and conclude the sampling strategies for IDS in section 2. IDSampling is proposed in details in section 3. In section 4 experiment results is presented, then we conclude and outline the direction of the future work in section 5.

2 Sampling Strategies for IDS

Usually there are less than 0.5% attack packets in network, while the figure soars when large-scale anomaly happens. For example, approximately 70% packets are attack packets in some DDOS. The possibility of discarding the attack packets is low in normal traffic, for most of the packets are clean. However, in this case if some attack packets do be discarded, the lost will influence detection rate greatly. When large-scale anomaly happens, the possibility of discarding the attack packets increases, but the traffic feature of high repetition accompanying with large-scale anomaly helps to capture attack packets. So different sampling strategies should be applied in different cases and we will discuss them separately as follows.

Large-scale anomaly bears some distinct traffic features, for example, the volume of packets and flows will soar and exhaust IDS soon. Another distinct feature is that the packets are of high repetition. So in this case it is efficient for the recovery method to make a reasonable approximation though multiplying the results by the reciprocal of the sampling rate. Experiment results of some researches^[8] have proven that if the sampling rate is too low and distorts the metrics heavily, the detection rate will be too

low to make sense. At this time the sampling strategy should concentrate on the most abnormal flows to guarantee the sampling rate of abnormal flows high enough to detect attacks and ignore the others for saving the limited resources.

Compared with the large-scale anomaly attack, the recovery method of other attacks can't be as simple as multiplying, for their behaviors are no longer of high repetition. These attacks conquer system via various vulnerabilities and we'll conclude the sampling strategy for them by analyzing their typical attack process.

A typical attack is composed of seven phases. ① shielding the source of the attack, ② collecting the information of victims, ③ exploiting the vulnerability, ④ breaking into the victim, ⑤ clearing the attack trace, ⑥ launching attack, ⑦ executing backdoor program. At the beginning ② and ③ are usually carried out by kinds of port scan, so most attack information lies in the short flow. With the attack evolving the long flow gets to contain more and more attack information. In steps ④~⑥ the attacker tend to attempt many different methods to maximize the success possibility. This kind of redundancy makes sampling method promising in detection. Sampling the short flow in high rate at the beginning will help to identify the attack flow as early as possible with the low cost, then we can increase the sampling rate of these attack flows after signing them, and discard the others without signs. As the percent of the attack flow is rather low in this case, the sampling strategy can undoubtedly reduce the pressure of IDS.

To summarize, IDSampling applies different strategies in different cases. ① When large-scale anomaly happens, IDSampling should focus on the most abnormal flows to guarantee the sampling rate of abnormal flows high enough to detect attacks. ② When large-scale anomaly doesn't happen, IDSampling should sample the short flow in high rate to guarantee to detect the attack and sign the flow in the beginning, then it can sample subsequent packets of the flow with signs in high possibility and discard the others in high possibility.

3 IDSampling Method

IDSampling profiles the traffic feature to apply different sampling strategy. So time is divided into measurement bins for the traffic statistic. Because the traffic feature is of self-similarity and lone-range dependence, it's reasonable to apply sampling on the statistic results of the former bin. As the processing method is same in all bins, the latter discuss will be limited in one. To make IDSampling clear, firstly we will introduce some important definitions.

Definition 1. feature_i represents a traffic feature, i.e, source IP or destination port.

Given an empirical histogram vector denoted as $\text{feature}_i = \{(x_i, n_i), i=1, 2, \dots, N\}$ which means that feature_i has N different values of x_i and each x_i occurs n_i times,

the entropy of the feature_i is defined as: $H(\text{Feature}_i) = -\sum_{i=1}^N \left(\frac{n_i}{S}\right) \log_2 \left(\frac{n_i}{S}\right)$, while

$S = \sum_{i=1}^N n_i$ is the total number of observations in the bin.

The entropy of the feature_i provides a convenient summary statistic for a feature distribution's tendency to be concentrated or dispersed which will help to identify abnormal flows. We focus on four feature entropies. Let $H(\text{srcIp})$, $H(\text{srcPort})$, $H(\text{dstIp})$ and $H(\text{dstPort})$ be the entropy of source address, source port, destination address and destination port. The value of them lies in the range $(0, \log_2 N)$. The figure takes on the value 0 when the distribution is maximally concentrated, i.e., all observations are the same. The feature entropy takes on the value $\log_2 N$ when the distribution is maximally distributed, i.e., $n_1=n_2=\dots=n_n$.

To tell a feature entropy is normal or not, we introduce the expectation variation of the feature entropy as following.

Definition 2. Given X as a random variable, $\exists E(X)$ and δ_X which stands for the expectation and the variance of X separately. Then expectation variance of X , denoted by ζ_X , is defined as $\zeta_X = \frac{|X - E(X)|}{\delta_X}$, which illustrates how far away X is deviated from $E(X)$.

For a feature entropy i , there are two thresholds ζ_{i_1} and ζ_{i_2} which can get by training data. If $\zeta_i \leq \zeta_{i_1}$, we can tell i is normal, if $\zeta_i > \zeta_{i_2}$, then we can tell i is abnormal, or $\zeta_{i_1} < \zeta_i < \zeta_{i_2}$, and we are uncertain whether or not i is normal.

The rest of this chapter is organized as follows. Section 3.1 explains how to adapt sampling rate due to IDS bottleneck. In section 3.2 IDSampling is introduced. Section 3.3 gives a detailed description of the single sampling method based on four traffic feature entropies which works when large-scale anomaly happens. In section 3.4 another complicated sampling instructed by the feedback of the detection results is applied by default. In section 3.5 the feedback methods are discussed. Finally performance analysis is provided in section 3.6.

3.1 Adapting Sampling Rate

The sampling rate is restricted by the bottleneck of IDS which lies in either CPU or memory. We suppose that the IDS manufacture will guarantee in its configuration that CPU will not be overwhelmed when memory is exhausted. So under such restriction whether or not the receiving buffer overflows is a signal to tell whether or not the processing rate of IDS can catch up with the input rate. And as long as the buffer overflows, the sampling method will be launched to enable IDS to work efficiently with limited resources. So the sampling rate is determined by equation (1).

$$P = \text{the packets processing rate of IDS/the arrival rate of network packets} \quad (1)$$

3.2 IDSampling

IDSampling is composed of three steps, and each step will be discussed in details in the following sections.

- ① In the beginning of the bin IDSampling counts the adaptive sampling rate P by equation(1) as discussed in 3.1, then tells whether or not large-scale traffic anomaly occurs using the method discussed in [6].If the answer is yes, then turn to ②, else turn to ③.
- ② For each arriving packet X , IDSampling applies the single sampling strategy based on four traffic feature entropies as discussed in section 3.3,
- ③ For each arriving packet X , IDSampling applies the complicated sampling strategy by default as discussed in section 3.4.

3.3 The Single Sampling Strategy of IDSampling

As discussed in section 2, IDSampling will capture the most “abnormal” traffic when large-scale anomalies happen. And the sampling strategy is called the single one based on the feature entropies.

Table 1 lists a set of anomalies commonly encountered in backbone network traffic and the changing tendency of the entropy’s distribution, in which “ $\uparrow$ ” means the feature is becoming more distributed, while “ $\downarrow$ ” means more concentrated and “-” means uncertainty. The change of the distribution tendency caused by the abnormal is obvious according to their definitions. It is shown in table 1 that each of the abnormal affects at lest two feature entropies, and that’s why the anomaly detection method^[6] based on the feature entropy is more accurate than those focus on traffic volume.

Table 1. Qualitative effects on the feature entropy by various anomalies

Abnormal Lable	Defination	H(srcIp)	H(srcPort)	H(dstIp)	H(dstPort)
Alpha Flows	Unusually large volume point to point flow	$\downarrow$	-	$\downarrow$	-
DOS/DDos	Denial of Service Attack(distributed or single-source)	$\uparrow$	-	$\downarrow$	-
Flash Crowd	Unusual burst of traffic to single destination, from a “typical” distribution of sources	-	$\uparrow$	$\downarrow$	-
Port Scan	Probes to many destination ports on a small set of destination address	-	-	$\downarrow$	$\uparrow$
Network Scan	Probes to many destination addresses on a small set of destination ports	-	-	$\uparrow$	$\downarrow$
Outage Events	Traffic shifts due to equipment failures or maintenance	$\downarrow$	-	$\downarrow$	-
Point to Multipoint	Traffic from single source to many destinations, e.g., content distribution	$\downarrow$	$\downarrow$	$\uparrow$	$\uparrow$
Worms	Scanning by worms for vulnerable hosts(special case of Network Scan)	-	-	$\downarrow$	$\uparrow$
				$\uparrow$	$\downarrow$

Large-scale abnormal traffic is of high frequency and some repetition pattern. And as the table 1 shows, the most concentrated feature entropy indicates the outstanding repetition pattern of the abnormal traffic as well as high frequency. So the flow that ranks the topN of the smallest feature entropy is the most “abnormal” traffic. The single sampling strategy is also composed of three steps:

- ① Choose the feature whose entropy is the smallest in four feature entropies. Denoted as $\text{feature}_{\text{smallest}} = \{(x_i, n_i), i=1, 2, \dots, R\}$.
- ② Sort $\text{feature}_{\text{smallest}}$ by n_i , and compute topN which satisfies equation (2).

$$\zeta_{H(\text{feature}_{\text{smallest}})} > \zeta_{i_2} \quad (2)$$

where $H(\text{feature}_i) = -\sum_{i=1}^N (\frac{n_i}{S}) \log_2 (\frac{n_i}{S})$, as defined in definition 1.

③ For an arriving Packet X, if $\text{feature}_{\text{smallest}}$ of X is included in topN, then samples X using packet sampling method at sampling rate P, else discards it.

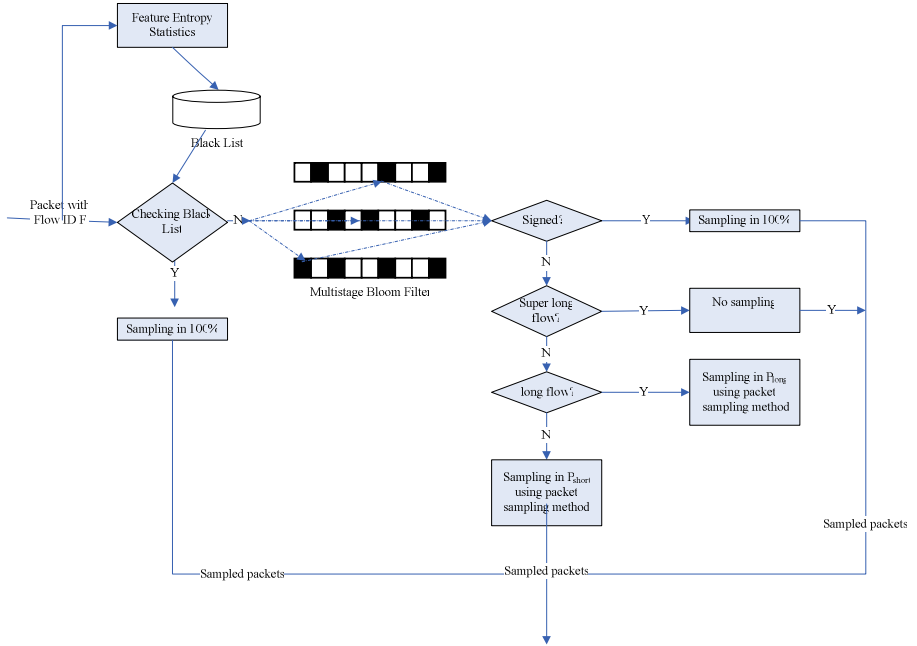


Fig. 1. The flow chart of the complicated sampling strategy

3.4 The Complicated Sampling Strategy of IDSampling

When large-scale anomaly doesn't occur, IDSampling will sample the coming packet with different sampling rate, and the rate is dependent on its flow length. To find a tradeoff between the efficiency and the consumption, we divide all flows into three types: the short flow whose length ≤ 10 , the long flow whose length ≤ 1000 and the super long flow whose length > 1000 . Let P_{short} , P_{long} and $P_{\text{superlong}}$ be the sampling rate of them respectively. W_{short} and W_{long} denote the priority of the short flow and that of the long. For the reasons listed in section 2, $P_{\text{superlong}}=0$ which means the super long flow will not be sampled any more, and P_{short} and P_{long} are determined by equation (3) and (4) as the following. In equation(3), P is the adaptive sampling rate determined by equation(1). We adopt notable multistage Bloom Filter to count the flow length^[10], and the method is not discussed here for brevity.

$$P_{\text{short}} + P_{\text{long}} = P \quad (3)$$

$$P_{\text{short}} / P_{\text{long}} = W_{\text{short}} / W_{\text{long}} \quad (4)$$

① For each arriving packet X, it will pass multistage Bloom Filter to tell which type of the flow it is belonged to.

② If X belongs to the flow which is signed as an attack flow, it will be sampled at 100% rate. If X belongs to the short flow, it will be sampled at high sampling rate P_{short} using packet sampling method. If X belongs to the long flow, it will be sampled at low sampling rate P_{long} using flow sampling method, or X will be discarded for it belongs to the super long flow.

Fig. 1 illustrates how the complicated sampling strategy works under the feedback of the following detection results.

3.5 The Feedback Method of IDSampling

As the sampling method is different with the flow length, the feedback method also varies with the flow length.

The feedback method of the long flow is to sign the counter of the Bloom Filter which works in the same way as counting the flow length. When a packet X is confirmed as an attack packet by the detection engineer, a hash on its flow ID is computed in every stage of the Bloom Filter and the corresponding counter is signed (in bolded as shown in Fig.1). Since all packets belonging to the same flow hash to the same counter, X will be confirmed as a subsequence of the attack flow if all the X's counters are signed and it will be sampled at rate 100%.

However, the aforementioned feedback method is absolutely useless in the short flow. Firstly, the last time of the short flow is very short, usually scales in several million seconds. So the short flow will be finished entirely before the hash feedback method begins to work. Secondly, the number of the short flow is too large to sign. And even if we can sign all the short flow efficiently, the error positive of the multistage Bloom Filter will be too high to work. Finally, IDS can not afford so much communication information between the detection module and the sampling module. Considering the fact that the attack in the short flow is of high repetition, so clustering information is efficient to capture such a characteristic. The reasonable feedback method of the short flow is that putting the topN of the feature_{smallest} into the blacklist when large-scale anomaly happens and sampling the packet which is included in the blacklist at 100% rate. The maintenance of the blacklist is neglected here.

3.6 Performance Analysis

The memory consumption of IDSampling is composed of two parts mainly. One part is the multistage Bloom Filter, while the other is the statistics of the traffic feature entropy. Denoted by b the number of hash counters of each Bloom Filter stage, let p and n be the number of the active packets and the flows, and N presents the number of topN flow which is decided by equation(2). Thus, the total memory of multistage Bloom Filter is $O(bd)$, and that of the statistics is $O(n)$. As the processing speed is concerned, the performance of IDSampling is dependent on the different sampling strategy. But in both cases the feature entropy statistics will costs $O(n)$ in the

beginning of each time bin. IDSampling will process each packet in $O(1)$ when the simply sampling strategy is applied, while it will cost $O(\log_2 N)$ in the complicated sampling case which is caused by checking the blacklist. And in this case IDSampling will calculate the topN which costs $O(n \log n)$, so the processing cost of the complicated sampling strategy is $O(n + n \log n)$ which is the sum of the two phases.

4 Experimental Results

Our experiments adopt Snort as IDS and DARPA 1999, a notable 5-week dataset for evaluating IDS, as the experiment data set. In fact, we only use the inside tcpdump files of the former 4 weeks. Firstly, IDSampling is trained by the files of the first and the third week to get different threshold, secondly it will be tested by the files of the fourth week to evaluate sampling accuracy. Before processed by IDSampling, the files are denoted by $Traffic_{all}$. And after processed by IDSampling, they are denoted by $Traffic_{sampled}$. Then we use Snort to detect $Traffic_{all}$ and $Traffic_{sampled}$ to get two different alert files respectively, denote by $Result_{all}$ and $Result_{sampling}$. Then $Result'_{sampling}$ and $Result'_{all}$ are generated from $Result_{sampling}$ and $Result_{all}$ separately after redundancy eliminating. The accuracy of IDSampling can be measured by $accuracy_{dection}$ which equals $Result'_{sampling} / Result'_{all}$. Another thing worthy to mention is that $accuracy_{dection}$ is lower than the actual figure that IDSampling achieves because of the limit detection ability of Snort. So compared to the labeled attack list, we will consider the attack is detected successfully if the amount of its sampling data is greater than the threshold.

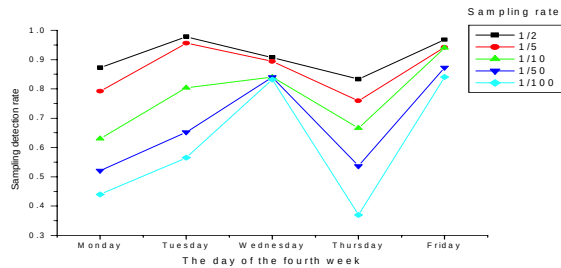


Fig. 2. The detection rate of IDSampling at different sampling rate of the fourth week

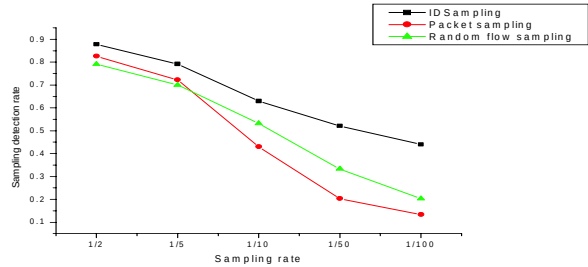


Fig. 3. The detection rate of three different sampling methods

Fig 3 plots the detection rate in every day of the 4th week at different sampling rate. It is shown that the detection rate descends monotonously with the sampling rate. The detection rate(D_r) is rather high at high sampling rate ($\leq 1/5$), the max D_r reaches 97.8% at 1/2 sampling rate. As the sampling rate drops, the D_r falls down monotonously, but the dropping speed varies in different day. The attenuation of D_r is quite large in the Monday, Tuesday and Thursday, while that of the Wednesday and the Friday varies little. For example, in the Tuesday the D_r drops from 97.8% at 1/2 sampling rate to 56.5% at 1/100 sampling rate, while the figure of the Friday is 96.8% and 84.1 respectively. By analyzing the labeled attack list, we confirm that attacks in the Monday, Tuesday and Thursday are quite small, usually scales less than 100 packets. So the attack is distorted heavily when the sampling rate drops to 1/100 and it will be missed entirely. However, there are a large-scale R2L attack in the Wednesday and a large-scale Probe attack in the Friday respectively. So the traffic repetition and the clustering information help to maintain D_r even in low sampling rate(1/100). To summarize, IDSampling scores pretty well at high sampling rate and it performs better in large-scale anomaly.

To evaluate how efficient IDSampling is, Fig 4 compares the D_r of it to that of the other two notable sampling methods, packet sampling and random flow sampling, in different sampling rate using the inside tcpdump file of the Monday as a trace. It is indicated that all three sampling methods seem to affect the detection in a similar manner, however, their relative impact on the degradation of D_r is quite different. IDSampling outperforms the other method in all sampling rates and it is the most robust one. It drops by 43.8% from the sampling rate 1/2 to 1/100, and the drops from the other two method are 69.2% and 58.8 separately. The D_r of packet sampling falls to 13.4% and that of random flow sampling is 20.3% when the sampling rate decreases to 1/100, while IDSampling still remains 44%. To summarize, IDSampling is the most efficient one for it is under the help of the feedback of the following detection result and other clustering information of the traffic feature entropy. So the D_r of IDSampling is pretty higher when sampling rate is low. The performance of the packets sampling stands the lowest and the random flow sampling lies in the middle. That's because the random flow sampling samples the flow without bias and so it can get more accurate information about IPs and ports than the packet sampling, but the packet sampling absolutely leans to the long flow.

5 Conclusion

In this paper we employ a novel sampling method IDSampling which samples abnormal packets with the help of the feature entropy when large-scale traffic anomaly occurs, and it incorporates dynamic feedback from the detection engine to further maximize the successful sampling ratio. To address the performance limits of IDS, it is a cost-effective yet scalable solution which can work in line with Gigabit links. The experiment results show that IDSampling is well suited to extract the attack packet and it is pretty nice in high sampling rate($<1/10$). However, if the sampling rate is relatively low compared with the attack scale, the detection rate of IDSampling will drop due to heavy metric distortion. In another word, IDSampling can not guarantee the performance of IDS.

Our ongoing work is centered on extending IDSampling to be a sampling with detection accuracy guarantees. In particular, we are studying additional information that can aid in better detecting anomalies by their root-cause, analyzing the mathematics model of it and investigating how much information is necessary for recovering an attack from the sampling data.

References

1. Bos, H., Huang, K.: Towards Software-Based Signature Detection for Intrusion Prevention on the Network Card. In: Valdes, A., Zamboni, D. (eds.) RAID 2005. LNCS, vol. 3858, pp. 102–123. Springer, Heidelberg (2006)
2. Cho, Y., Mangione-Smith, W.: Fast reconfiguring deep packet filter for 1+gigabit network. In: IEEE Symposium on Field-Programmable Custom Com[puting Machines (FCCM), NaPa, CA (April 2005)
3. Fanklin, R., Caraver, D., Hutchings, B.: Assisting network intrusion detection with reconfigurable hardware. In: Proceedings from filed Programmable Custom Computing Machines (2002)
4. Chen, X.x., fang, B.x.: The architecture of Intrusion detection system in high-speed network. Computer research development, [J] 41(9), 1481–1487 (2004)
5. Charitakis, I., Anagnostakis, K., Markatos, E.: An active traffic splitter architecture for intrusion detection. In: Proceedings of 11th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS 2003), Orlando, October 2003, pp. 238–241 (2003)
6. Lakhina, A., Crovella, M., Diot, C.: Mining Anomalies Using Traffic Feature Distributions. In: Proc. ACM SIGCOMM 2005, Philadelphia, PA, USA (August 2005)
7. J.MAI, ,SRIDHARDAN, A., Chuah, C.N, Aang, H., Impack of packet sampling on portscan detection. IEEE Journal on Selected Areas in Communication(2006).
8. Mai, J., Chuah, C.N., Sridharan, A., Ye, T., Zang, H.: Is sampled data sufficient for anomaly detection? In: Proc. of the 6th ACM SIGCOMM on Internet measurement, Brazil (2006)
9. Brauckhoff, D., Tellenbach, B., Wagner, A.: Impact of packet sampling on anomaly detection metrics. In: Proc. ACM SIGCOMM 2006, Rio de Janeiro, Brazil (2006)
10. Estan, C., Varghese, G.: New Directions in Traffic Measurement and Accounting[C]. In: SIGCOMM 2002, pp. 270–313 (August 2003)

Fast Traffic Classification in High Speed Networks

Rentao Gu, Minhuo Hong, Hongxiang Wang, and Yuefeng Ji

Key Laboratory of Optical Communication and Lightwave Technologies

Beijing University of Posts and Telecommunications

Haidian District, Beijing 100876, China

{jamesgrt, hongmh1984}@gmail.com, {wanghx, jyf}@bupt.edu.cn

Abstract. A novel approach for fast traffic classification in the high speed networks is proposed, which bases on the protocol behavior statistical features. The frame lengths, arrival times and direction of packets are collected from the real data flows. Comparing the features of the unknown flow with the protocol masks, we can judge which application protocol this flow belongs to. Distinct from other statistic methods, we use the “universal flow-based inter-arrival time” to overcome the influence of RTT variance so that a set of excellent protocol masks is site-independent and time-independent. Because there is no need for character string searching and complex algorithms, the proposed approach can be easily deployed in the hardware of high speed network equipments.

Keywords: Communication system traffic, measurement system data handling, networks, protocol.

1 Introduction

Accurate network traffic classification is fundamental to numerous network activities. However, only a few methods could meet the strict requirements of high speed networks: fast, accurate and online [1]. Currently, application classification practices rely to a large extent on the use of the port numbers and packet payload examination [2]. And at the time, the method based on packet payload examination [3] is scarcely an option because of complexity limitations, privacy issues and payload encryption by the applications. There is another kind of classification method, which uses the statistical properties of network traffic to find out the protocol behavior characters [4].

The approach shown by Crotti et al. [5], focuses on the statistical analysis of network traffic and shows promising results for fine-grained protocol classification. Although this technique tries to express the statistical properties of each application layer protocol with simple size/inter-arrival time/direction distributions, it is different than the one presented in this paper. Their approach is based on the traditional definition of “inter-arrival time”, protocol fingerprint and anomaly score, so that the protocol fingerprints are site-dependent and time-dependent. The presented approach is based on the concept of “universal flow-based inter-arrival time (UFIAT)”, in which the transmission time is eliminated. So it can counter the RTT variance and make the protocols masks site-independent and time-independent.

2 The Methodology

In this section, we present our traffic classification method based on UFIAT. There are three main steps to complete the traffic classification: collecting information, building the protocol masks and making judgments. Comparing the features of the unknown flow with the known protocol masks, we can judge the application protocol type of this flow.

Without loss of generality, suppose $\mathbf{F}$ is a data flow between a client (connection originator) and a server (connection recipient), composed of at least $(N + 1)$ IP packets, from P_1 to P_{N+1} , where N is the number of the packets under our observation, $\mathbf{F} = \{P_1 P_2 P_3 \dots P_i \dots\}$.

Where P_i represents i -th packet which arrives the measurement point in the bidirectional flow. T_i is the arrival time of the packet P_i at the measurement point, as shown in Fig.1.

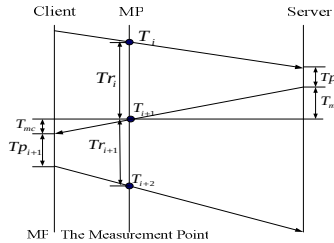


Fig. 1. The relationship between protocol process time, transmission time, and the flow based inter-arrival time

Frame length L_i : The length of the i -th frame.

Protocol processing time Tp_i : The protocol processing time between the i -th packet and the $(i+1)$ -th packet. The protocol processing time represents the actual processing time of the application layer protocol, which is a most distinct feature for each application protocol.

Flow based Inter-arrival time (FIAT) Tr_i : The difference between the arrival times of two successive packets at the measurement point. It is based on bidirectional measurement. $Tr_i = T_{i+1} - T_i$

Transmission time: It is divided into two kinds: the transmission time between the measurement point and the client T_{mc} , and the transmission time between the measurement point and the server T_{ms} . Both of them are protocol-independent.

Universal flow-based inter-arrival time (UFIAT) Tu_i : It is used to estimate the value of protocol processing time. From the formula Tr_i , we obtain the UFIAT:

$$Tu_i = \begin{cases} Tr_i - 2 * T_{ms} & (P_i : Client \rightarrow Server \text{ and } P_{i+1} : Server \rightarrow Client) \\ Tr_i - 2 * T_{mc} & (P_i : Server \rightarrow Client \text{ and } P_{i+1} : Client \rightarrow Server) \\ Tr_i & (others) \end{cases} \quad (1)$$

Note that the value of T_{ui} relies on the directions of the i -th packet and the $(i+1)$ -th packet, as shown in Fig.1, in which the transmission time variance is eliminated. The variance of transmission time has great influence on FIAT just like inter-arrival time. If the influence is reduced, the protocol masks we derive will be much better. That's why we use UFIAT in our protocol masks instead of FIAT.

Our protocol masks require the information of the frame length and universal flow-based inter-arrival time. The packet length is easy to get. According to (1), Tu_i depend on T_i , T_{mc} and T_{ms} . Among these three parameters, the measurement of T_{mc} and T_{ms} seems a little difficult. Here, we use three-way handshake to obtain the initial value of T_{mc} and T_{ms} . Because the set-up process of TCP connection need little time, the protocol process time Tp_1 and Tp_2 can be ignored. Usually, the first three packets of the TCP flow are "SYN", "SYN ACK", "ACK" packets. And then, we can get $T_{ms} = 0.5 \times Tr_1 = 0.5 \times (T_2 - T_1)$ and $T_{mc} = 0.5 \times Tr_2 = 0.5 \times (T_3 - T_2)$. Where T_1, T_2, T_3 are the arrival times of these three packets at the measurement point respectively.

Then each flow F can be characterized as an ordered sequence of N pairs $F_i = \{L_i, Tu_i\}$, ($1 \leq i \leq N$). Because each protocol has its own process scheme, it is reasonable to classify the traffic according to the distribution of F_i . The generation of the protocol masks starts from the set of Probability Density Matrix (PDM_i), estimated from a set of flows generated by the same known protocol.

A basic building block of decision method is the value that i -th component of the PDM_i assumes in P_i . In fact, the value of $PDM_i(P_i)$ in (L_i, Tu_i) represent the possibility that the pair P_i is set to (L_i, Tu_i) . And the value $PDM_i(P_i)$ gives the correlation between the i -th packet of the unknown flow and application layer protocol described by the specific PDM used. The higher value means the higher possibility that the flow was generated by such protocol. To avoid the "noise" influence, the judgment can be competed according to several PDM_i .

3 Analysis and Experimental Results

Experimentally, we validated the classification techniques with traffic traces collected in the campus network of BUPT. The measurement points were two edge routers at different places and we used mirror port to collect the network traffic.

The hit ratios of the method based on UFIAT are higher than those of the methods based on traditional inter-arrival time, as shown is Table 1. The more important issue is that the masks based on UFIAT can be built according to one node and be used for another one. Because UFIAT-based methods don't rely on the RTT distribution, it is not necessary to rebuild the masks when the popular sites change. In one word, a set of perfect protocol masks based on UFIAT could be used universally at any time and any place. The results also show that the filter applying to the protocol masks will bring much benefit to the hit ratios.

Table 1. The hit ratio comparison between the masks based on UFIAT and traditional inter-arrival time with and without gauss filter

	UFIAT with filter	Inter-arrival time with filter	UFIAT without filter	Inter-arrival time without filter
HTTP	99.55%	96.69%	88.90%	86.90%
POP3	99.02%	94.65%	43.72%	26.79%
FTP	95.63%	81.25%	93.75%	89.58%
POP3S	99.84%	98.63%	63.64%	12.64%
Streaming	97.74%	84.28%	82.98%	40.15%

4 Conclusion

In this paper we propose a novel approach for traffic classification. To make this approach be site-independent and time-independent, the concept of “universal flow based inter-arrival time” (UFIAT) and a method to measure the transmission time using TCP hand-shakes are introduced. These ideas can be used for both of the traditional application and encrypted traffic. The results show that the approach performs better than the methods based on traditional inter-arrival time, especially with gauss filter. The feature of is due to the elimination of the transmission time variance.

Acknowledgments. This research was supported by the NSFC (No. 60572021, 60711140087, No. 90704006), National 973 Program (No. 2007CB310705), National 863 Program (No. 2006AA01Z238), PCSIRT (No.IRT0609), ISTCP (No.2006DFA11040), 111 Project (No. B07005), P. R. China.

References

1. Zhang, G., Xie, G., Yang, J., Min, Y., Zhou, Z., Duan, X.: Accurate Online Traffic Classification with Multi-Phases Identification Methodology. In: 2008 Proc. of IEEE Consumer Communications and Networking Conference, pp. 141–146 (2008)
2. Karagiannis, T., Broido, A., Brownlee, N., Claffy, K., Faloutsos, M.: Is P2P dying or just hiding? In: GLOBECOM 2004: Proc. Of IEEE Global Telecommunications Conference, Riverside, USA, pp. 1532–1538 (2004)
3. Broder, A., Mitzenmacher, M.: Network applications of bloom filters: a survey. *Internet Mathematics* 1(4), 485–509 (2003)
4. Karagiannis, T., Papagiannaki, K., Faloutsos, M.: BLINC: multilevel traffic classification in the dark. In: SIGCOMM 2005: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications, pp. 229–240 (2005)
5. Crotti, M., Gringoli, F., Pelosato, P., Salgarelli, L.: A statistical approach to IP-level classification of network traffic. In: 2006 Proc. IEEE International Conference on Communications, pp. 170–176 (2006)

Service Impact Analysis Framework Using Service Model for Integrated Service Resource Management of NGN Services

Seung-Hee Han, Bom-Soo Kim, Chan-Kyou Hwang, and Jae-Jin Lee

Internet Research Department, KT Network Technology Laboratory,
463-1, Jeonmin-dong, Yuseong-gu, Daejeon, Korea
{shehan, kbsjmj, ckhwang, jaejin}@kt.com

Abstract. Service Delivery Platform (SDP) is proposed to create various applications more easily for Next Generation Network (NGN) services. This trend is a challenge for telecommunication operators who will manage not only existing services but also new NGN services such as multimedia services and converged services. We present a service model to analyze service impact for NGN services and service impact analysis framework using the service model.

Keywords: Service model, Service impact analysis, Service Delivery Platform, Service Oriented Architecture, Service resource management.

1 Introduction

As various multimedia and converged services are getting more important in NGN, systems which provide those services should be managed more systematically along with network infrastructure [1]. Currently, the network infrastructure is managed efficiently by various powerful NMS and SMS. On the other hand, operators face some challenges when managing service resources overall in application layer.

Definition of NGN services helps us have insight into NGN service management. In the scope of NGN, services provide added value to network connectivity. Such services are usually composed of a set of applications hosted in one or more application server in interaction with databases, relying on a system infrastructure and delivered to the customer via a networking infrastructure. Specially, the services are defined as a set of reusable service component, which may be belonging to one or more services [2]. The definition of NGN services gives the necessity of operation management system to manage various applications, application servers, system infrastructure, and network infrastructure overall. We focus on the service layer of NGN, because we have no overall solution in the service layer now.

SDP is proposed to create various applications of NGN services more easily in the service layer [3]. Service Oriented Architecture (SOA)-based SDP is suggested because the concept of SDP corresponds with the concept of SOA, which creates a new service by composing the existing services [4]. This paper focuses on operation management system for the SOA-based SDP environments. The paper is organized as

follows: In Section 2, we explain the characteristics of SOA-based SDP. We discuss a service model for NGN services executing in SOA-based SDP in Section 3. After then service impact analysis framework using the service model is also proposed. Finally, we conclude this paper in Section 4.

2 SOA-Based SDP

SDP is proposed to implement such NGN services efficiently. The common components of SDP are selected for general adaptation of our proposal and depicted in the left side of Figure 1. An example of the corresponding structure of SOA-based SDP is depicted in the right side of Figure 1 [4]. **Service Execution Environment** consists of service enablers, common services and service orchestration environment. Service enablers have functions which link to IMS Core or legacy network and provide also basic telecom services-presence, call control, messaging and so on. Common services provide basic services like e-mail and VOD to reuse when creating new value added services. If we build SDP with SOA concept, the right side of Figure 1 shows an overview of the SDP implemented by SOA solution.

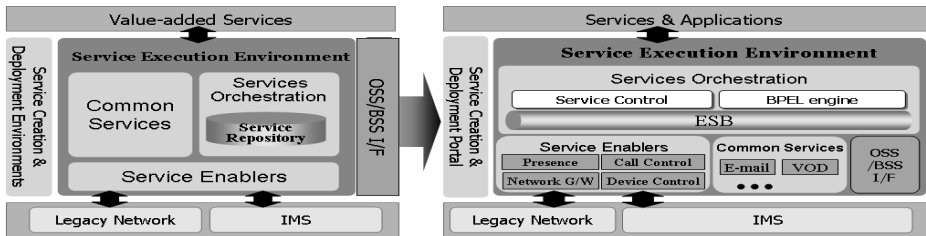


Fig. 1. Common structure of SDP and an example of corresponding structure SOA-based SDP

3 Service Impact Analysis

Service impact analysis shows the correlation among services which compose the service monitored and service resources. We can find some root causes of a fault and also recognize a service affected by some faults of resources through the service impact analysis. A service model is used for the service impact analysis. We abstract the correlations among service resources and model them in UML.

3.1 Service Model for SOA-Based SDP

A service model represents the structure of a service. A service model has three main components [5]: service topology representation, measurement topology representation, and state representation. A Service is composed of three categories of resources: logical resources, physical resources and network resources. They are drawn from the Application, Computing and Network domains, and include, for example, software, IT Systems and Network Elements.

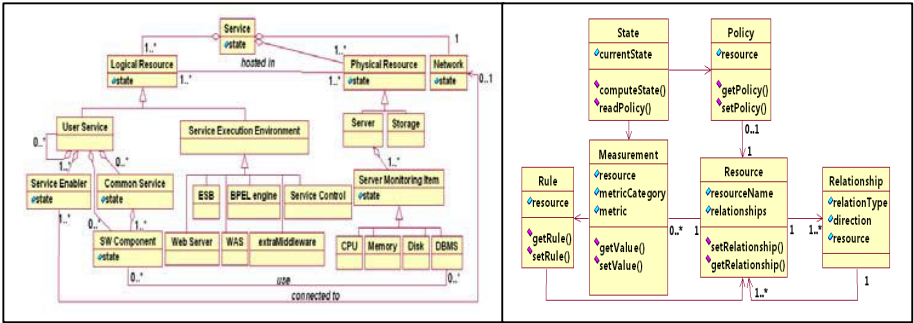


Fig. 2. Service model of NGN services (the left side) and a class diagram for service measurement topology and service state representation (the right side)

Service topology representation of the proposed service model is depicted in the left side of Figure 2. This model is drawn from the viewpoint of the service resource management in the application layer. A service state can be computed from the states of logical resources, the states of physical resources and the state of network. These relationships are expressed by aggregation relationships between the whole, i.e. the *Service* class and three parts, the *Logical Resource* class, the *Physical Resource* class and the *Network* class. The *Network* class abstracts network layers to provide the service. The *Physical Resource* class generalizes equipment such as application servers on which applications are running. The *Logical Resource* class generalizes non-physical components from SOA-base SDP architecture and legacy system: composite services (user service), common services, service enablers, ESB, BPEL engine and so on. The *State* class, the *Measurement* class and the *Policy* class (the right side of Figure 2) are derived for measurement topology and state representation.

3.2 Service Impact Analysis Framework Using the Service Model

The framework for the service impact analysis is depicted in Figure 3. The service model realization part in Figure 3 performs mapping between the service model saved in repository and the service resources configuration information. After then, correlations between resources are saved into SADB. The SADB stores information of all service resources and configurations.

The service state computation part in Figure 3 carries out computation of the state of service resources and analysis by the requests of the Service Impact Analyzer. First, the Service Resource State Calculator brings the correlation information between service resources from SADB(a) via the Integration Manager(b). The Measurement Performer inquires the metrics and the rules to compute its state(c). The correspondent information is taken from the *logical resource* and the *physical resource performance and event information database*(d) by the Integration Manager(e, f). The State Computation Policy Manager reads the policies to analyze the resource(h) and sends them to the Service Resource State Calculator(i).

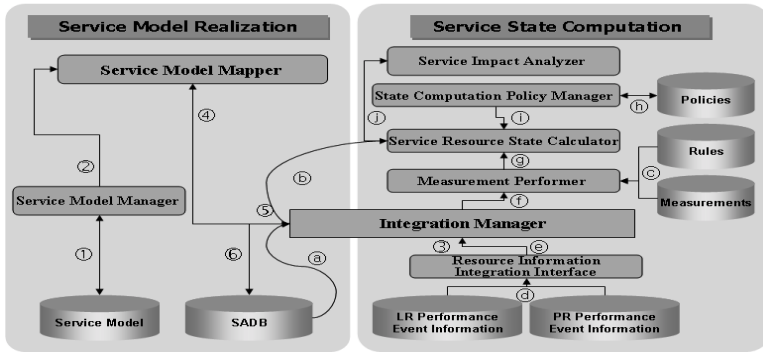


Fig. 3. Service impact analysis framework, LR is an abbreviation for logical resource and PR is an abbreviation for physical resource

The Service Resource State Calculator computes the state of each service resource from values which the Measurement Performer gives (8). The state is escalated by traversing the relationships of the realized service model to the Service Impact Analyzer (11). The Service Impact Analyzer shows the results and relationships between resources through GUI (graphical user interface).

4 Conclusion and Future Work

This paper discussed the service impact analysis framework in complex environments of SDP and legacy system. First, the service model for service resource management was drawn from the structure of SOA-based SDP. After then service impact analysis framework using the service model was described. In the future the service model will be refined by developing prototype of service resource management system.

References

- 1 Bang, Y.H.: NOM 2.0: Innovative Network Operations and Management for Business Agility. IEEE, Communications Magazine (2008)
- 2 Yahia, I.G.B., Bertin, E., Deschrevel, J.P., Crespi, N.: Service Definition for Next Generation Networks. In: IEEE Proc. of the Intern. Conf. on Networking, Intern. Conf. on Systems and Intern. Conf. on Mobile Communications and Learning Technologies (2006)
- 3 Ohnishi, H., Yamato, Y., Kaneko, M., Moriya, T., Hirano, M.: Service Delivery Platform for Telecom-Enterprise-Internet Combined Services. In: IEEE Global Telecom. Conf. (2007)
- 4 Maes, S.H.: Service Delivery Platforms as IT Realization of OMA Service Environment-Service Oriented Architectures for Telecommunications. In: IEEE Wireless Communications and Networking Conf. (2007)
- 5 Caswell, D., Ramanathan, S.: Using Service Models for Management of Internet Services. IEEE Journal on Selected Areas in Communications 18(5) (2000)

A Trusted Quality of Web Services Management Framework Based on Six Dimensional QoWS Model and End-to-End Monitoring^{*}

Nan Guo, Tianhan Gao, and Bin Zhang

Information and Science Engineering College, Northeastern University, China. 110004
{guonan, gaoth, zhangbin}@ise.neu.edu.cn

Abstract. To present a trusted QoWS management framework, we firstly propose a six dimensional QoWS model which includes expected QoWS, agreed QoWS, delivered QoWS, perceived QoWS, transmitted QoWS, and statistic QoWS to represent the various requirements of the QoS property and describe QoWS comprehensively and objectively. Then, a QoWS manager is introduced to the legacy SOA to enhance the quality ensurence. It adopts Simple Network Management Protocol to respectively capture delivered QoWS and perceived QoWS to obtain the end-to-end performance of services. Thus it makes a comprehensive assessment of statistic QoWS representing the long-term quality of the given services.

Keywords: Quality of Web Services, QoS Model, End-to-End Monitoring.

1 Introduction

In the presence of multiple Web services with overlapping or identical functionality, service consumers need objective quality of web services (QoWS) criteria to distinguish one service from another. The current researches have provided effective mechanisms to manage QoWS in an objective and reliable way [1-4]. They either obtain supplied QoWS by providers or evaluate QoWS mainly on the client feedback. The on-going researches are not sufficient to represent the trusted performance of services under dynamic environment. In this paper, a six dimensional QoWS model is firstly presented to describe the various requirements of the same quality property. Then a new style QoWS management model is presented to obtain the reliable end-to-end performance of the transactions of Web services.

2 Six Dimensional QoWS Model

To present a trusted QoWS management model, an objective QoWS model is required firstly. The on-going work regard quality criterion as one dimension which means it

^{*} Supported by NNSF of China (No. 60773218).

represents the only one required value of some quality criterion. Under such circumstances, we can hardly assess the quality of the Web service on one dimensional criterion without analyzing the comprehensive performance. Additionally the probability attributes such as availability, reliability, accessibility and reputation are the statistics of quality collected by the trusted party and can be neither claimed by providers nor active monitored.

Thus, a six dimensional QoWS model is proposed. It is a comprehensive model for the Web service life cycle, including selection, agreement, monitoring, and assessment. QoWS is classified to six dimensions as follows:

- Expected QoWS (EQoWS): it is oriented by service selection and represents the required or restricted quality from consumers;
- Agreed QoWS (AQoWS): it is oriented by service agreement and represents the agreed service level and SLA parameters;
- Delivered QoWS (DQoWS): it is oriented by service monitoring and represents the quality supplied by providers;
- Perceived QoWS (PQoWS): it is oriented by service monitoring and represents consumers' experience of service performance;
- Transmitted QoWS (TQoWS): it is oriented by service monitoring and represents the network performance;
- Statistic QoWS (SQoWS): it is oriented by service assessment and represents the long-term quality of services. It is aggregated from DQoWS and PQoWS periodically.

Let us make an example of the most typical quality parameter - *response time* to show the scenario. The *response time* is typically defined as the amount of time between sending the request and receiving a response and computed using the processing time and the transmission time. In six-dimensional QoWS framework, the *response time* in agreed QoWS is the agreed one signed by provider and consumer before invoking the service. The one in delivered QoWS is monitored and mainly includes the service execution duration and server system processing duration. The one in perceived QoWS is monitored mainly includes the round trip delay of network and *delivered response time*. The one in transmitted QoWS is as the same meaning to the network delay. The one in statistic QoWS is the respective statistics of *delivered response time*, *perceived response time* and *transmitted response time*. Thus it is clear that the single dimension of *response time* is not able to represent the objective and realistic QoWS.

3 A Trusted QoWS Management Framework

On the base of six-dimensional QoWS model, a trusted QoWS management framework is proposed to respectively get six dimensional quality criterions. From the view of SOA, it acts as the fourth role cooperating with provider, consumer and registry. The Fig.1 illustrates the extended SOA and the proposed QoWS management framework.

In a Web service community, service providers publish their WSDL in the unified service registry and declare SLA in the QoWS manager. Service consumers send

requests to the service broker of the service community to implement service discovery and selection. Before invoking services, providers and consumers need to engage in QoS negotiation and sign SLA. It specifies what an end user wants and what a provider is committing to provide on the context of specifying one service level.

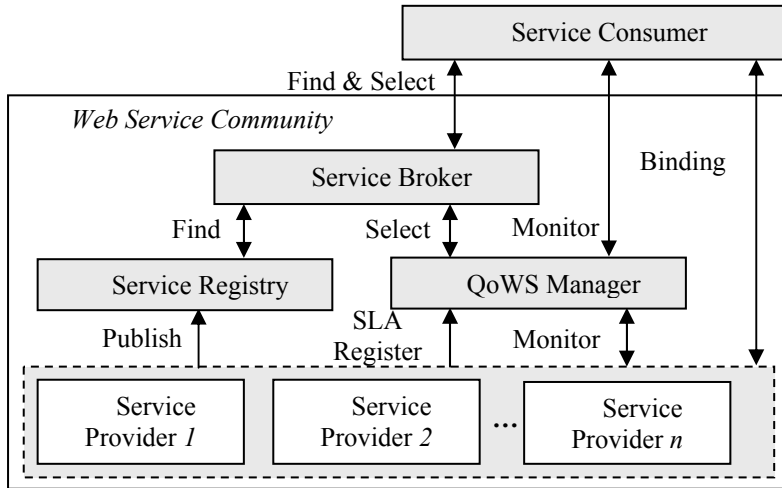


Fig. 1. The extended service oriented architecture and QoWS management framework

The QoWS properties of Web services have the same meaning to the management objects in the network and system management. We explore the idea of applying the Simple Network Management Protocol (SNMP) and technologies to our QoWS management framework. SNMP has the capability of managing services through creating specified QoWS MIB, adopting manager/agent model for end-to-end monitoring and SNMP protocol for transaction.

We design two QoWS MIB groups. SQoWS Group describes the management information of servers. It includes two tables, *servTable* and *sessionTable*. *servTable* records the all services' description supplied by the provider. *sessionTable* records the all sessions executed by the services. A session is identified by *sessionID* and represented by *servURL* and *clientAddr*. It has four statuses that are *active*, *delivered*, *failure* and *collected* to respectively indicate if the service is running to process the requests, has returned the response, was not available in the *validTime*, or has been monitored by QoWS managers. If QoWS manager finds a non *collected* session, it will get the *delivered parameters*.

Another QoWS MIB group is CQoWS Group. It describes the all sessions invoked by the client. As the same to the *sessionTable* in SQoWS, a session is identified by *sessionID*. The QoWS manager respectively collects the management information of server and client and generates the end-to-end performance of service by picking the original data of the same session identified by the same *sessionID*. A session also has four statuses that are *active*, *perceived*, *failure* and *collected* to respectively indicate if the request is being processing, has received the response, has not received the

response in the *validTime*, or has been monitored by QoWS managers. If QoWS manager finds a non *collected* session, it will get the *perceived parameters*.

Agents work at both provider side and consumer side to capture the original data of the performance of services. The QoWS Manager works as managers to collect the original data periodically and generate the respective values of six dimensional quality properties of the given service transaction. To assess the given service over a period of time, it periodically collects the service’s delivered QoWS and perceived QoWS and computes the statistic QoWS, which is depicted as Table 1. The QoWS list gives customers the comprehensive profile of service quality. It is oriented to a service operation and the service level it belongs to. All parameters in one service level are described in both agreed QoWS and statistic QoWS. In agreed QoWS, any parameter is either delivered one or perceived one, which is indicated by D/P property. It is important to compute the reputation, which is an indicator of the trustworthiness of providers in complying with agreed QoWS or consumer’s perception in complying with agreed QoWS.

The key role of the proposed six dimensional QoWS model is providing the trusted reference to service selection. Before service consumers bind a service to invoke the required operation, they will check the QoWS lists of the candidate services to get the claimed service levels and the relative parameters of agreed QoWS and statistic QoWS. Through making match of agreed QoWS, statistic QoWS with expected QoWS consumers constrict, they can select the right one.

Table 1. The QoWS List

Operation	Service Level		Agreed QoWS		Statistic QoWS	
Operation ID	Level i	Parameter j	D/P	Value j	Value j	Reputation
						
						

References

1. Liu, Y., Ngu, A., Zeng, L.: QoS Computation and Policing in Dynamic Web Service Selection. In: Proceedings of the 13th International World Wide Web Conference, pp. 66–73 (2004)
2. Maximilien, E.M., Singh, P.P.: Toward Autonomic Web Services Trust and Selection. In: Proceedings of the 2nd International Conference on Service Oriented Computing, pp. 212–221 (2004)
3. Jurca, R., Faltings, P., Binder, P.: Reliable QoS Monitoring Based on Client Feedback. In: Proceedings of the 16th International Conference on World Wide Web, pp. 1003–1012 (2007)
4. Erradi, A., Maheshwari, P.: wsBus: QoS-aware Middleware for Reliable Web Services Interactions. In: Proceedings of the IEEE International Conference on e-Technology, e-Commerce and e-Service, pp. 634–639 (2005)

Least Interference Optimization Based Dynamic Multi-path Routing Algorithm in ASON

Tong Zhao, Yueming Lu, and Yuefeng Ji

Key Laboratory of Optical Communication and Lightwave Technologies
Ministry of Education, Beijing University of Posts and Telecommunications,
100876, Beijing China
zhaotongbupt@gmail.com, {ym lu, jyf}@bupt.edu.cn

Abstract. Based on current load balance and minimum interference routing technology, a least interference optimization based dynamic multi-path routing algorithm (LI-DMA) in automatically switched optical network (ASON) is proposed. LI-DMA employs least interference technology and takes future demands into consideration while calculating routing paths. In LI-DMA, the link cost is dynamically adjusted according to its residual capacity and criticality, and the traffic is distributed to several disjoint paths equally while loading the connections. Simulation results show that, compared with conventional routing algorithms, LI-DMA can efficiently decrease the blocking probability of network and achieve better performance of load balance.

Keywords: ASON, routing, interference, load balance.

1 Introduction

With enduing traditional optical transport network with unprecedented flexibility and expansibility, ASON represented the future direction of next generation optical networks. The Control Plane of ASON would be based on the Generalized Multi-Protocol Label Switching (GMPLS). There are many advanced routing algorithms on GMPLS that are being researched. Dynamic Traffic Sharing (DTS) algorithm is QoS based routing algorithm [1]. Least Interference Optimization Algorithm (LIOA) is traffic based routing algorithm based on Minimum Interference Routing Algorithm (MIRA), and the interference is defined by the number of flows carried by a link [2]. It has been proved that a reasonable and efficient multi-path routing could decrease the blocking probability of the network.

In this paper, we propose a least interference optimization based dynamic multi-path routing algorithm in ASON. The proposed algorithm LI-DMA refers to LIOA and takes advantages of multi-path routing algorithm to support traffic engineering and load balancing. In LI-DMA, the link cost consists of link criticality and link load cost, and average multi-path load-sharing strategy is employed. Simulation results show that, compared with conventional routing algorithms, LI-DMA can efficiently decrease blocking probability of network and achieve better performance of load balance.

2 The Proposed Algorithm

First, two crucial components involved in the algorithm, namely, the definition of Link cost function and multi-path load-sharing strategy, are provided. Then, the executing procedure of LI-DMA is introduced in detail.

2.1 Definition of Link Cost Function

In LI-DMA, consider a link criticality function W_C as the product of the number of flows carried on the link N_C and the length of the link D . Link load rate P_L is defined as the ratio of used bandwidth of the link B_U and total bandwidth of the link B_T . We provide the definition of link load function W_L as the following shows:

$$W_L = \frac{P_L^\alpha}{1 - P_L^\alpha} * N_C * D = \frac{P_L^\alpha}{1 - P_L^\alpha} * W_C \quad (\alpha \geq 1) \quad (1)$$

Where α is a parameter representing a tradeoff between P_L and W_L . In this paper, let $\alpha = 4$. It can be concluded from formula (1) that, the value of W_L is less than that of W_C when link load rate P_L is relatively low. However, W_L increases exponentially with P_L and it outweighs W_C as P_L increases.

At last, link cost function W consists of link criticality function W_C and link load cost function W_L :

$$W = W_C + W_L = (1 + \frac{P_L^4}{1 - P_L^4}) * N_C * D \quad (2)$$

It is crystal-clear that when link load rate is low, link criticality W_C has a dominating influence on link cost W , and thus LI-DMA is characterized by high throughput which is inherited from LIOA. With the link load rate P_L increasing, link load function W_L increases exponentially and link cost now mainly depends on W_L .

2.2 Multi-path Load-Sharing Strategy

LI-DMA employs average multi-path load-sharing strategy, and it spreads traffic in a Traffic Group (TG) over disjoint paths equally while loading connections. TG is a set of traffic that has the same attributes, for example, the same source and destination, the same priority, the same resource attribute, the same routing constraints, the same survival strategy, etc. A TG has a parameter S_n and a parameter V_p to index maximum path sharing number and virtual path respectively. Besides, another essential parameter vp_index is introduced in TG which points to current V_p . V_p is calculated first and each connection is distributed in disjoint V_p equally while loading connections in a TG where V_p is often set to 2 or 3.

2.3 The Executing Procedure

Note that all of steps introduced below are implemented in the source node.

- Step 1. Source node searches a valid TG for the connection when it receives a connection request. If fails to find a valid TG for this connection, source node will create a new TG and attach this connection to the new TG.
- Step 2. Source node verifies whether the current number of V_p in TG is less than S_n . If so, source node will try to calculate another V_p according to formula (2). If source node calculates a new V_p successfully, it will set vp_index to the new V_p and go to step 6. Otherwise, go to step 3.
- Step 3. Point vp_index to the next V_p in TG.
- Step 4. Judge whether current V_p can satisfy requirement of the connection. If current V_p can meet the needs of the connection, go to step 6. Otherwise, go to step 5.
- Step 5. Check whether every V_p in TG is traversed. If not, go to step 3. Otherwise, source node will deny the connection request.
- Step 6. Signaling module is called according to current V_p .

3 Case Study

Cases of static traffic and dynamic traffic are studied in this section and each case applied three routing schemes. Scheme 1 adopts SPF algorithm, and scheme 2 uses DTS algorithm. Scheme 3 employs LI-DMA and we simulate it with three different maximum path sharing number.

3.1 Routing Simulation of Static Traffic

Topology of NSF network is used. This network includes 14 nodes and 800 links. 400 connections, 750 connections and 1000 connections are respectively distributed in the network to simulate the scenario of network with light load, heavy load and over load.

Under the scenario of network with light load, compared with SPF algorithm, LI-DMA can decrease variance of link load rate by 40% averagely. However, LI-DMA achieves this at a cost that it increases the mean value of link load rate.

Compared with SPF, both DTS and LI-DMA increase utilization ratio of network cost significantly. Besides, LI-DMA with multi-path load-sharing strategy consumes more network cost than DTS does. Especially when the network is with heavy load, compared with DTS, LI-DMA has expense of an extra average 11% in utilization rate of network cost.

It is noteworthy that LI-DMA($S_n=3$) achieves higher utilization rate of network cost than LI-DMA($S_n=2$) does under the scenarios of network with heavy load and over load. This is because LI-DMA($S_n=2$) can maintain lower blocking probability and carry more traffic under these scenarios, therefore, LI-DMA($S_n=2$) occupies more network resource than LI-DMA($S_n=3$) does.

3.2 Routing Simulation of Dynamic Traffic

Assume the traffic is Poisson flow, and connection requests follow Poisson distribution with an average arrival rate β . Traffic duration matches negative exponential distribution with mean value $1/\lambda$. Then, the traffic intensity ρ is the ratio of β and λ .

As shown in Table 1, SPF results in highest blocking probability regardless of traffic intensity while LI-DMA(Sn=2) leads to lowest blocking probability under various traffic intensity. LI-DMA leads to lower blocking probability than DTS does, which is due to LI-DMA uses least interference technology and takes future demands into consideration while calculating routing paths. Table 1 also reveals that LI-DMA(Sn=2) results in lower blocking probability than LI-DMA(Sn=3) does. This is because more traffic in LI-DMA(Sn=3) selects indirect path and consumes more link resource than LI-DMA(Sn=2) does.

Table 1. Results of Blocking Probability under Different Traffic Intensity

ρ (Erl) \ Scheme	SPF	DTS	LI-DMA (Sn=1)	LI-DMA (Sn=2)	LI-DMA (Sn=3)
100	0.197	0.001	0.001	0	0
200	0.318	0.007	0.006	0.002	0.002
400	0.421	0.178	0.139	0.113	0.114
800	0.583	0.537	0.516	0.493	0.502

4 Conclusions

LI-DMA can maintain high throughput which is inherited from LIOA and prevent high-loaded link from being occupied as possible. With average multi-path load-sharing strategy, LI-DMA makes network load more balanced.

It can also be concluded that it is proper to select a relatively low maximum path sharing number when the traffic intensity of network is great.

Acknowledgments. This research was supported by the National 863 Program (No. 2007AA01Z252), NSFC (No.60572021, 60711140087, 90704006), National 973 Program (No. 2007CB310705), ISTCP (No.2006DFA11040), 111 Project (No. B07005), NCET (06-0090), PCSIRT (No.IRT0609), P. R. China.

References

- 1 Hui, L., Yueming, L., Yuefeng, J.: Dynamic Traffic Sharing Algorithms for Balanced Optical Internet. In: Proceeding of SPIE APOC 2005, p. 60221S (2005)
- 2 Bagula, B., Botha, M., Krzesinski, E.: Online Traffic Engineering: the Least Interference Optimization Algorithm. In: Proceeding of IEEE ICC 2004, pp. 123–1236 (2004)

QoS-Aware Scheduling in Emerging Novel Optical Wireless Integrated Networks

Min Luo, Hui Li, Yueming Lu, and Yuefeng Ji

Key Laboratory of Optical Communication and Lightwave Technology,
Ministry of Education, Beijing University of Posts and Telecommunications,
Beijing, 100876, P.R. China
buptluomin@gmail.com, {lihui, ymlu, jyf}@bupt.edu.cn

Abstract. As the era of converging digital industry takes shape, to make full use of huge bandwidth offered by fiber and mobile characteristics, the integration of wireless and optical networks is a potential solution for increasing capacity and mobility as well as decreasing costs in the access network. In this paper we present the basic architecture of an emerging integration of EPON and hybrid wireless networks, and propose an efficient scheduling to deliver diverse services by taking the quality of service (QoS) requirement into consideration. The simulation results demonstrate that our scheme is capable of collecting the QoS requirement information throughout the network, and the process facilitates the reduction of network transmission overhead.

Keywords: FMC, EPON, WiMAX, WLAN, 3G, QoS.

1 Introduction

To make full use of huge bandwidth offered by fiber and mobile characteristics presented via wireless scheme, the integration of wireless and optical networks is a potential solution for increasing capacity and mobility as well as decreasing costs in the access network [1], [2], [3].

In this paper, we present the novel optical wireless integrated networks which consist of EPON and Hybrid Wireless Networks, then propose an efficient scheduling to deliver diverse services by taking the quality of service (QoS) requirement into consideration. Performance evaluation are presented and compared in Section 3. Finally, we conclude the paper in section 4.

2 Novel Optical Wireless Integrated Architecture: The Integration of EPON and Hybrid Wireless Networks

In this paper, the optical wireless integration uses a cellular network UMTS and two wireless technologies WLAN, WiMAX with a fiber backbone IP network.

2.1 Proposed Network Architecture

Figure.1 shows the network architecture that wired and hybrid wireless network are integrated to achieve access platforms integration. ONU1 and ONU3 carry out FTTx solution to provide high reliability and extend reach. ONU2 receives signals from Access Point (AP) of WLAN, Node B of 3G and Base Station of Wimax, and then sends them to the Central Office (CO). The signal processing is concentrated in the CO. The scenario is that a wireless subscriber having a smart.

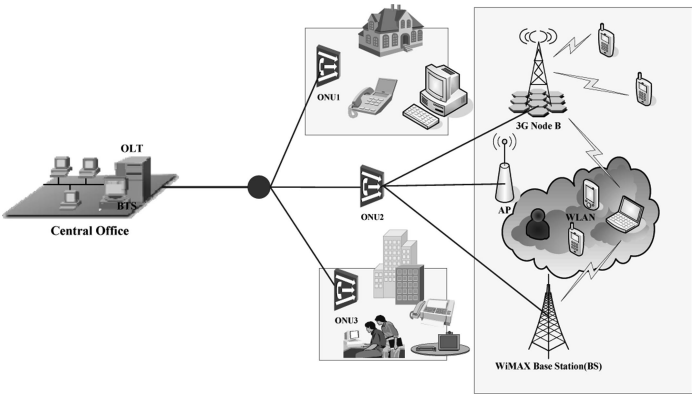


Fig. 1. Architecture of optical and wireless network over EPON

2.2 Design of QoS-Aware Scheduling

A QoS-Aware scheduling is developed to keep the same channel between wired and wireless networks by using mobile devices. The mobile device determines the nature of application and starts the process of scheduling. The operation flowchart is shown in Figure.2. Mobile devices select network interface according to the application in use,

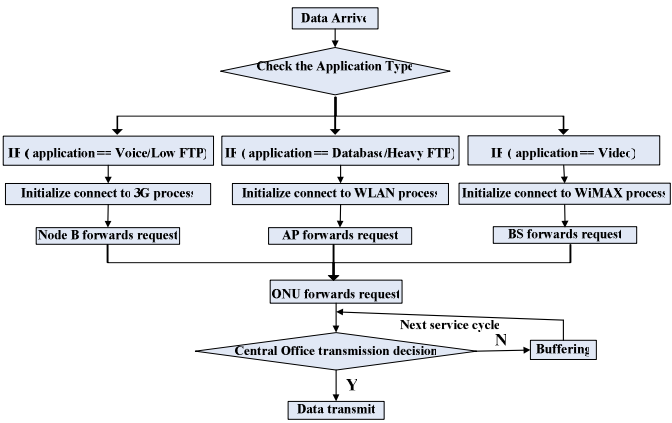


Fig. 2. QoS-aware scheduling operation flowchart

i.e. voice and HTTP data uses 3G cellular network, where as for heavy FTP, databases and online banking can use secure WLAN and for real time video calling, conferencing and live games WiMAX is prominent candidate supporting up to 70Mbps. Instead of each hop making the transmission decision in a distributed way based on its local information of network resources, the central office collects the resource information and service request from all of its users, and then manages the network resources centrally under the guideline of QoS provisioning.

3 Performance Evaluation

In this section, performance of our QoS-Aware scheduling proposal is evaluated by simulations. In the simulation network model, the server site supported different application traffic such as Voice, HTTP, Heavy FTP, Databases and real time video conferencing [4]. All the simulation network traffic parameters for both scenarios are described in Table 1.

Table 1. Network data traffic definition for scenarios

WiMAX/WLAN and 3G over EPON	APPLICATION				
	H=heavy	L=light			
	Database	HTTP	FTP	Voice	Real Time Video Conf.
No Scheduling	H	L	H	PCM	H
QoS-Aware Scheduling	H	L	H	PCM	H

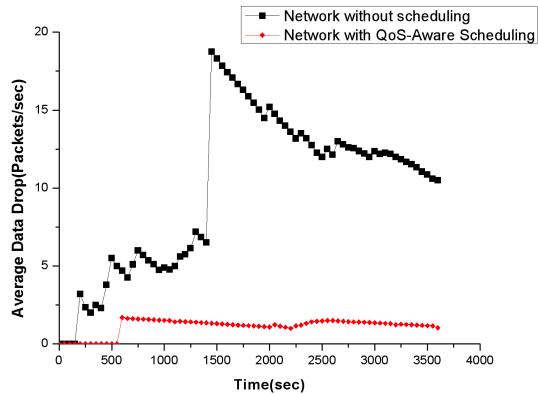


Fig. 3. Network end-to-end packet drop

Figure 3 shows the obvious difference between the peak-to-peak network packet drops. It reached 18 packets drop/second and gradually degraded to an average of 10 packets per second. On the other hand, QoS-Aware scheduling raised the system performance by reducing the packet drop up to 60%. This on demand adaptation of new interface reduces a great number of data drop.

4 Conclusion

In this paper, QoS-Aware scheduling for the Integration of EPON and Hybrid Wireless Networks are successfully simulated and discussed. QoS-Aware scheduling not only enhanced the network overall performance but also the reliability and reduction of packet drop up to 50%, which is a very enormous achievement for any real and time sensitive applications. Our future work will be on the pursuit of intra-class scheduling and service fairness. The research on queue size and data loss is also of high interest to the management of integrated optical wireless networks.

Acknowledgments. This research was supported by National 973 Program (No. 2007CB310705), the NSFC (No. 60572021, No. 90704006), National 863 Program (No. 2006AA01Z243), NCET (06-0090), PCSIRT (No.IRT0609), ISTCP (No.2006DFA11040), 111 Project (No. B07005), P. R. China.

References

- 1 Sudhir Dixit: Hybrid Optical-Wireless Networks. Technical report, the Optical Fiber Communication Conference and Exposition (2007)
- 2 Luo, Y., Wang, T., Weinstein, S., Cvijetic, M., Nakamura, S.: Integrating optical and wireless services in the access network. Technical report, the National Fiber Optic Engineers Conference (2006)
- 3 Christian, T., Hassnaa, M., Houda, L.: Hybrid Wireless Networks: Applications, Architectures and New Perspectives. In: 3rd Annual IEEE Communications Society, vol. 3, pp. 848–853 (2006)
- 4 Chaudhry, S.R., Al-Raweshidy, H.S.: An Application Controlled Handover for Heterogeneous Radio in Wired Wireless Integrated Networks. In: Mobile and Wireless Communications Summit, pp. 1–4 (2007)

An Energy-Efficiency Route Protocol for MIMO-Based Wireless Sensor Networks

Qing-Hua Wang^{*}, Yu-Gui Qu, Zhi-Ting Lin, and Bao-Hua Zhao

Department of Electronic Engineering and Information Science,
University of Science and Technology of China, P.O. Box 230027, Anhui, China
flyer@mail.ustc.edu.cn

Abstract. A novel multihop energy-efficiency routing protocol for MIMO-based WSN is proposed in this paper. The protocol integrates cooperative MIMO technology with the multihop routing and energy consumption. When building route, the node not only consider its own energy consumption, but also consider the cost to the base station of the next hop node, thus the end-to-end energy consumption is decreased. Simulation results indicate that this protocol can obviously reduces the networks energy consumption.

Keywords: Wireless Sensor Networks, cooperative MIMO, multiple hops.

1 Introduction

Cooperative multiple-input multiple-output (MIMO) technology in WSN could reduce the transmission energy and improves the energy efficiency. The MIMO technology in WSN has been studied intensively [1-2]. Some researchers have propose different routing protocols for virtual MIMO-based WSN^[3,4]. However, energy consumption is not considered in these protocols.

In view of the shortcomings of existing routing protocols for WSN based on cooperative MIMO, an energy-efficiency MIMO routing protocol (EMRP) is proposed in this paper. The transmission energy consumption, remaining energy, and the cost of the next hop node to the BS are jointly considered to maximize the energy efficiency when routing in the new protocol. This routing strategy not only saves total energy consumption of the entire networks, but also distributes the energy consumption to all the nodes averagely.

2 Energy Consumption Model

The system model of multihop cooperative MIMO protocol is shown in Figure 1. Some nodes group together as a cluster to form virtual antenna arrays and transmit data cooperatively. Among the nodes which constitute a cluster, one is central node (CN),

^{*} This paper is supported by the National Natural Science Foundation of China under Grant No. 60602016, HUAWAI Foundation No.YJCB2007061WL and Innovation Fund for Graduate Student of USTC under Grant No. KD2005042.

the remaining are cooperative members (CMs)^[5]. The transmission in each hop includes two operations: the CN broadcasts the data to its CMs and then the cooperative nodes encode and transmit the data to the CN in the next hop.

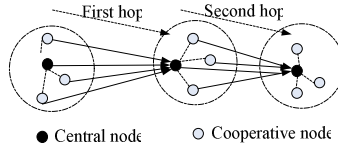


Fig. 1. The sketch of multihop cooperative MIMO routing protocol

The energy consumption in each hop includes two parts: the energy spent on transmitting to next hop by cooperative MIMO and that spent on the broadcasting of CNs to their CMs. The latter is consumed in local cluster and is much less than that spent on inter-cluster communication, having little influence on routing.

MQAM is used as the modulation scheme. The number of cooperative nodes in a cluster is M_t . The MAC protocol proposed in [4] is used. According to this MAC protocol, the channel state information (CSI) is available at the transmitters. Hence the nodes could adjust the transmission power respectively on the basis of equal SNR.

The average energy consumption per bit when cluster C_{N_t} transmits to the CN N_m in the next hop by cooperative MIMO can be approximated as^[3]

$$E_{bt}(C_{N_t}, N_m) = \frac{2}{3}(1+\alpha)\left(\frac{bP_b}{4}\right)^{-\frac{1}{M_t}} \frac{2^b-1}{b} \sum_{N_i \in C_{N_t}} \frac{(4\pi)^2 d_{N_t-N_m}^2}{G_t G_r \lambda^2} M_t N_0 N_f + \frac{M_t P_{ct} + P_{cr}}{R_b} \quad (1)$$

Where α is the efficiency of the power amplifier, G_t and G_r are the transmitter and receiver antenna gains respectively. N_0 is the single-sided noise power spectral density, P_{ct} and P_{cr} are the transmit and receive circuit power consumptions.

N_m transmits with transmit power P_{mt} to N_i , then the receival power at node N_i is

$$P_{ir} = P_{mt} G_t G_r \lambda^2 / ((4\pi)^2 d_{N_t-N_m}^2 M_t N_f) \quad (2)$$

Define $F(N_m, N_i)$ and C as:

$$F(N_m, N_i) = \frac{P_{mt}}{P_{ir}} = \frac{(4\pi)^2 d_{N_t-N_m}^2 M_t N_f}{G_t G_r \lambda^2} \quad C = \frac{2}{3}(1+\alpha)\left(\frac{bP_b}{4}\right)^{-\frac{1}{M_t}} \frac{2^b-1}{b} N_0 \quad (3)$$

Then Eq (1) can be reformulated as

$$E_{bt}(C_{N_t}, N_m) = C \sum_{N_i \in C_{N_t}} F(N_m, N_i) + (M_t P_{ct} + P_{cr}) / R_b \quad (4)$$

In order to save energy, Cluster C_{N_t} could choose the right node N_m which meets the $E_{bt}(C_{N_t}, N_m)$ is minimal among all the neighbor central nodes as its next hop.

3 Protocol Design

The BS first broadcasts routing establishment message (REM) with power P_{out} .

REM carries the following information: the BS ID, the hop number to the BS as well as the cost function field which indicates the cost from the current node to the BS.

On receiving this REM, each neighbor node first examines the hop number field of the REM. If the hop number in REM is smaller than the node's hop number, the REM is valid to this node, otherwise, the REM will be discarded.

If node N_j receives a valid REM from a CN N_s , N_j first calculates $F(N_j, N_s)$ according to (3). If N_j is not a CN, it will send the value to its CN N_c . Assume N_c has M central node candidates among which one appropriate node will be chosen as the next hop node. Utilizing all the $F(N_j, N_{s_i})$ calculated by its cooperative nodes N_j when communicating with different candidates N_{s_i} , N_c calculates all the $E_{bt}(C_{N_c}, N_{s_i})$ s. Here $N_{s_i} \in Set_{np}(N_c)$, $Set_{np}(N_c) = \{N_{s_1}, N_{s_2}, \dots, N_{s_M}\}$, is the set composed by M next hop candidates of N_c . The energy consumption metric when cluster C_{N_c} communicates with N_{s_i} is defined as:

$$Metric(C_{N_c}, N_{s_i}) = rE_{bt}(C_{N_c}, N_{s_i}) / E_{left}(C_{N_{s_i}}) + (1-r)Cost(C_{N_{s_i}}) \quad (5)$$

where $E_{left}(C_{N_{s_i}})$ denotes the minimal remaining energy value of the cooperative nodes in cluster $C_{N_{s_i}}$. r is weighting factor, $0 < r < 1$. The former part of the metric is on the basic of considering the energy consumption when communicating with the CN in the next hop and the minimal remaining energy of the cluster. The latter part of the metric is on the basic of considering the cost to the BS of the CN in the next hop. In addition, we could adjust the proportions of the two parts occupied in the metric by adjusting the value of r . Consequently we could determine the proportions of the communication cost of this current hop and the cost to the BS of the CN in the next hop node. The routing metric not only takes the energy consumption and the remaining energy into consideration, but also takes the total cost to the BS of the chosen path into consideration.

The CN N_c picks up the node N^* whose $Metric(C_{N_c}, N_{s_i})$ is the minimum in $Set_{np}(N_c)$ as its next MIMO hop node, namely

$$N^* = \arg \min_{N_{s_i} \in Set_{np}(N_c)} Metric(C_{N_c}, N_{s_i}) \quad (6)$$

After choosing the next hop node, node N_c will calculates its cost function value which is got by adding the cost function value of N^* to its the energy consumption per bit when communicating with node N^* , namely

$$Cost(C_{N_c}) = Cost(C_{N^*}) + E_{bt}(C_{N_c}, N_{N^*}) \quad (7)$$

Then the node N_c adjusts its transmit power which is used when sending data to the CN N^* by cooperative MIMO. Then N_c stores its routing table which contains the next hop node and the cost function value. After finishing the above operations, the CN N_c

broadcasts this routing result to its CMs. Then each CM N_i calculates its transmit power which is used to communicate with the CN N^* by cooperative MIMO.

The REM continues to broadcast till all the central nodes in the networks establish their cooperative MIMO routing.

4 Simulation Results

We compare the average energy consumption per hop of EMRP to VMRP in literature [4]. The simulation result is shown in figure 2. Some parameters in the simulation are set as follows: $M_t = 4$, $P_{ct} = 98.2\text{mW}$, $P_{ct} = 112.6\text{mW}$, $b=4$.

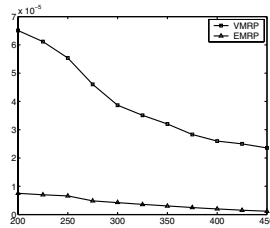


Fig. 2. The average energy consumption per hop of two protocols

In figure 2, the x-axis expresses the total number of nodes, the y-axis expresses the average energy consumption per hop per bit. Figure 2 shows that the average energy consumption of EMRP is far less than that of VMRP. The reason is that when routing in EMRP the node not only consider its own communication energy consumption, but also takes the cost to the base station of the next hop node into account, thus the end-to-end energy consumption is decreased. However VMRP doesn't consider the energy consumption when routing.

5 Conclusion

A novel energy-efficiency routing protocol is proposed for WSN based on cooperative MIMO technology in this paper. The cooperative MIMO technology is jointly considered with the multiple hops in this protocol. The energy consumption metric is defined for energy efficiency. When routing, the node not only considers itself transmission energy and the minimal remaining energy of the chosen next hop cluster, but also considers the cost to the BS. Simulations show that this protocol could save the total energy of the entire networks.

References

- 1 Cui, S., Goldsmith, A.J., Bahai, A.: Energy-efficiency of MIMO and Cooperative MIMO Techniques in Sensor Networks. *IEEE Journal on Selected Areas of Communications* 22(6), 1089–1098 (2004)

- 2 Li, X., Chen, M., Liu, W.: Cooperative Transmissions in Wireless Sensor Networks with Imperfect Synchronization. *IEEE Signal Process. Lett.* 12, 948–951 (2004)
- 3 Yuan, Y., Zhihai, H.: Virtual MIMO-Based Cross-Layer Design for Wireless Sensor Networks. *IEEE Transcation on Vehicular Technology* 55(3) (May 2006)
- 4 Gentian, J., Srikanth, V., Michalis, F.: A Framework for Distributed Spatio-temporal Communications in Mobile Ad Hoc Networks. In: *Proceedings of Infocom*. (May 2006)
- 5 Li, X., Chen, M., Liu, W.: Application of STBC-encoded cooperative transmissions in wireless sensor networks. *IEEE Signal Processing. Lett.* 12(2), 134–137 (2005)

Service Oriented T-MPLS Resilience Algorithm with Multi-QoS Constrained

Hua Qu^{1,2}, Ji-hong Zhao^{1,2}, and Hong-bao Mao²

¹ Xi'an Jiangtong University

² Xi'an Institute of Posts & Telecommunications,
Xi'an, China
qh@pub.xaonline.com

Abstract. In T-MPLS network, Resilience mechanism is needed for better network performance, however resilience methods related before couldn't match well all of those special and real requirements and face new challenges. In this paper, we discuss traditional resilience methods and analysis some limitations, Based on this, a new T-MPLS resilience algorithm based on Multi-service, Multi-recovery-patterns, Multi-QoS constrained is proposed, and the definition and mathematic model of QoS parameters around is expounded. Further more, in order to assure the success of establishing recovery path, a routing method based on Multi-QoS thresholds constrained is proposed.

Keywords: Transport MPLS, survivability, QoS, constrained routing.

1 Introduction

Even a short failure of a network link can cause substantial consequences, Resilience against failure is critical to today's network. Transport MPLS (T-MPLS^[1]) is a connection-oriented carrier-class transport for converge optical/packet network based on MPLS, it can provides managed point-to-point connections to different packet client layer networks. Since T-MPLS networks are expected to offer various services with QoS guarantees, it is a key issue to introduce resilience mechanisms in the networks. However, the traditional resilience methods^[2] have some disadvantages, such as only one recovery model, fewer QoS constrained, etc, it faces challenges: (1) How to be compatible with multiple recovery models^[3]; (2) How to fulfill the comprehensive assessment of QoS; (3) How to assure the success of establishing the best recovery path. A new T-MPLS Resilience algorithm based on Multi-service, Multi-recovery-patterns, Multi-QoS constrained (RMMM) is proposed in this paper. The proposed algorithm not only supports multi-recovery patterns and differentiated multi-service, but also is constrained by multi-QoS-threshold routing mechanism, and it improves transport performance of certain service.

2 Framework of the RMMM Algorithms

The RMMM algorithm of T-MPLS based on three recovery models, defines four QoS parameters, and adopts routing mechanism of multi-QoS threshold constrained to

choose the best recovery model and the best recovery path for different services. The general framework of RMMM algorithm can be divided into three stages (see Fig.1.): (1) Protection Switch/Merge LSR (PSL/PML) is preconfigured. (2) Based on QoS and node configuration, we can confirm protection level and thresholds of QoS parameters, farther select the best recovery model. (3) Based on QoS threshold and recovery model, we select the best recovery path and implement traffic forwarding.

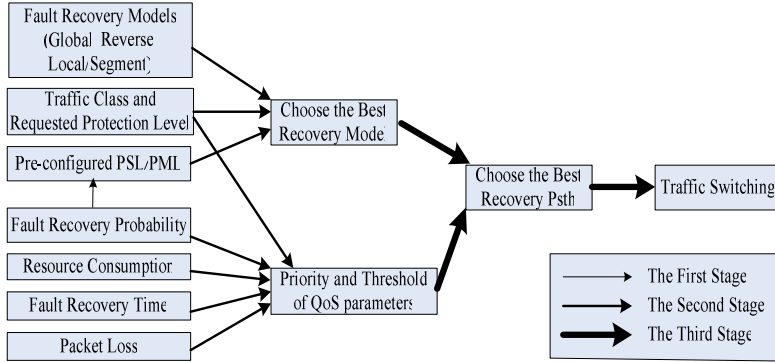


Fig. 1. The designed framework of RMMM

3 Principle of the Algorithm

According to Diff-Serv model defined by IETF, the traffic flow through T-MPLS networks is divided into three levels: Expedited Forwarding (EF), Assured Forwarding (AF) and Best Effort (BE). The QoS parameters defined in algorithm is as following:

(1) Recovery Probability (RP). Here, H is the hops in LSP, q_i is the refused probabilities for resource requirement, q_{ni} is the statistical failure probabilities, q_{ij} is the statistical failure probabilities. The RP of the whole LSP can be calculated:

$$RP = \prod_{i=1}^{(H+1)} (1-q_i) (1-q_{ni}) \prod_{j=1}^H (1-q_{ij}) \quad (1)$$

By considering of hypothesis that refused probabilities for resource requirement are equal, which is $q_i=q$, the statistical failure probabilities of any LSR and link are equal, that is $q_{ni}=q_n$, $q_{ij}=q_l$. So formula (1) is transformed into the following:

$$RP = (1-q)^{(H+1)} (1-q_n)^{(H+1)} (1-q_l)^H \quad (2)$$

Under the condition of $q \ll 1$, $q_n \ll 1$, $q_l \ll 1$, formula (2) is transformed into:

$$RP = 1 - (H+1)(q+q_n) - H \cdot q_l \quad (3)$$

(2) Recovery Time (RT). Here, H_r is the hops between failure point and PSL; t_{NOTi} is the time for processing FIS signal; t_{Ri} and t_{Ai} separately denote processing time for resource request and response; L_{Di} denotes transmit delay of the i^{th} link in working

LSP; LD_{ri} denotes transmit delay of the i th link in recovery LSP; α is a switch-order ($\alpha=1$ denotes local/segment recovery, otherwise $\alpha=0$); Q is a constant. Then:

$$RT \square (1-\alpha) \left(\sum_{i=1}^{H_r+1} t_{NOTi} + \sum_{i=1}^{H_r} LD_i \right) + \sum_{i=1}^{H+1} (t_{Ri} + t_{Ai}) + \sum_{i=1}^H LD_{ri} + Q \cdot \quad (4)$$

Suppose that delay of all LSRs is equal, which is $t_{NOTi}=t_{Ri}=t_{Ai}=t$, then

$$RT \square (H_r - \alpha \cdot H_r + 2H - \alpha + 3)t + (1-\alpha) \sum_{i=1}^{H_r} LD_i + \sum_{i=1}^H LD_{ri} + Q \cdot \quad (5)$$

(3) Packet Loss (N_{PL}): Packet Loss is related with certain recovery model. For reverse and local/segment recovery model, the challenge is N_{PL} in these three periods that are fault detection/location, recovery path establishment and traffic flow switchover, moreover, besides three periods, we need considering fault notification period in global recovery. The N_{PL} during recovery cycle is calculated as following:

$$N_{PL} = [(1-\beta) \left(\sum_{i=1}^{H_r+1} t_{NOTi} + \sum_{i=1}^{H_r} LD_i \right) + \sum_{i=1}^{H+1} (t_{Ri} + t_{Ai}) + \sum_{i=1}^H LD_{ri} + T_{DET} + T_0 + T_{SW}] R \cdot \quad (6)$$

Here, β is a switch-order, $\beta=1$ denotes reverse or local/segment recovery, otherwise $\beta=0$; T_{SW} is traffic switchover time and R is forward velocity.

Supposed signal process for all LSRs is equal, namely, $t_{NOTi}=t_{Ri}=t_{Ai}=t$, then

$$N_{PL} = \{(1-\beta)[(H_r+1)t + \sum_{i=1}^H LD_i] + 2(H+1)t + \sum_{i=1}^H LD_{ri} + T_{DET} + T_0 + T_{SW}\} R \cdot \quad (7)$$

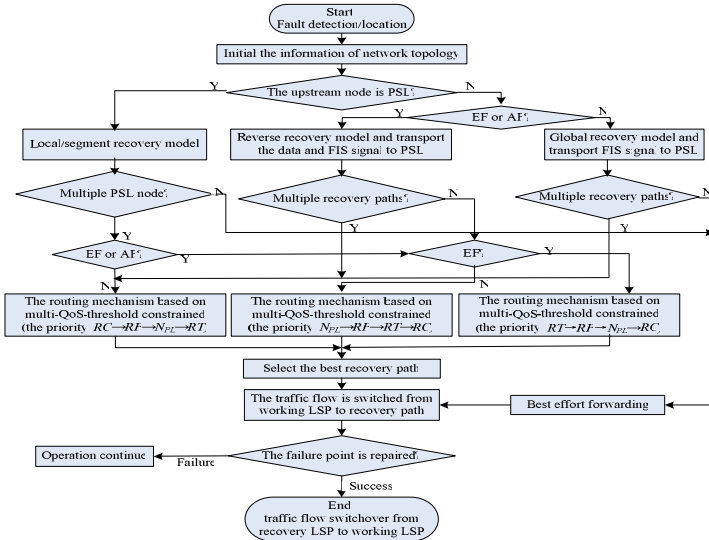


Fig. 2. The process of RMMM

(4) Resource Consumption (RC): RC is defined as bandwidth occupying or hops of path in traditional Resilience algorithm. Supposed link resource is denoted by bandwidth occupying. Here, D is the request bandwidth, C_i is equipment pay and maintenance cost, γ is a switch-order, $\gamma=1$ denotes reverse recovery, otherwise $\gamma=0$. So RC is denoted as:

$$RC = (H + \gamma \cdot H_r) d + \left(\sum_{i=1}^{H+1} C_i + \gamma \sum_{j=1}^{H_r+1} C_j \right) \cdot \quad (8)$$

Supposed the cost of general LSRs is equal, (that is $C_i=C$), and the cost of PSL/PML is $n \cdot C$. So formula (8) can be simplified as following:

$$RC = (H + \gamma \cdot H_r) d + (H + n\gamma \cdot H_r + n\gamma + 1) C \cdot \quad (9)$$

In brief, the above four variables will be the main proof of PSL/PML configuring, recovery model choosing and recovery path choosing.

When fault occurs, the process of RMMM is as following, see Fig.2.

4 Conclusion

T-MPLS is main tread of the next-generation packet transport networks. The resilience mechanism of telecommunication is the most top technique. RMMM inherits the restrictive idea in traditional recovery technique and mends their disadvantages; moreover, it meets the service need of differentiated QoS to a larger extent. It is of great importance for enhancing the survivability and robustness of T-MPLS network.

References

1. ITU-T Rec G.8110.1. Architecture of transport MPLS (T-MPLS) layer network (February 2006)
2. Jihong, Z., Hua, Q.: GMPLS Resilience algorithm Based on Constrained. Journal of UFST of China 34(1), 101–104 (2005)
3. Sharma, V., Crane, B.M., Makam, S., et al.: Framework for Multi-Protocol Label Switching (MPLS)-based Recovery. IETF RFC3496 (February 2003)

Appendix

The research work of this article is supported by the subsidy of National Foundation Science of China (sanction number.600572134, project classification: A, discipline code: F010106), and the subsidy of the Development Plan of National High-tech Research (863 Program) (the serial number of subject: 2007AA01Z254).

Network Stability Analysis Techniques Using the Virtual Articulation Node

Yonghyoun Kim, Kisu Kim, Sengjin Ahn, and Jinwook Chung

Dept. of Electrical and Computer Engineering, Sungkyunkwan Univ., Suwon, Korea
{yhkim, kskim, sjahn, jwchung}@songgang.skku.ac.kr

Abstract. This study proposes a method that measures the stability of networks by converting nodes with higher priorities into virtual articulation nodes. Further, although they're not articulation nodes, by considering critical nodes such as routers connected to external networks or servers that assume vital operations to networks, we are able to improve the monitoring of major nodes in networks and reallocate those nodes when designing networks in order to remove potential risk factors. We will elaborate on the method of creating virtual articulation nodes using virtual nodes and apply them to network topologies to perform a stability analysis through simulation.

Keywords: Articulation node, virtual node, virtual articulation node, network stability, NARI Node.

1 Introduction

Administrators need to manage networks in a stable and efficient way by using different functions of various protocol types and devices so that they are suitable to the network. Before going into such application management method, this study proposes a stable network management method through the structural design of networks.

The structure of a network can be illustrated with a connected undirected graph with nodes and edges [1]. The most important targets to monitor in the graph are articulation nodes, which are the only connections between node groups. Since they are located in critical points in a network through which all data must trespass, they were used as an element to measure network stability as proposed in this study. However, a network is not consisted of only articulation nodes. In other words, in order to measure network stability, degrees of importance must be tagged to nodes that are not articulation nodes.

This study proposes a network stability measurement method that applies the articulation nodes and adjacent trees suggested in previous studies to networks expanded with virtual articulation nodes generated by tagging importance levels to virtual nodes to nodes other than articulation nodes [3].

2 Related Studies

2.1 Articulation Node in Network

The structure of a network can be expressed with a connection undirected graph. Also, the factors of a broken point can be applied to some nodes where they connect

biconnected components. In other words, since this is the only path that connects two or more networks and the only node that can share data, it must handle more traffic than other nodes [1][2].

2.2 Network Stability Measurement Method Using Articulation Nodes and Adjacent Trees

This method measures the stability of a network by using the number of adjacent trees generated when an articulation node has been disconnected. Here, the articulation node represents the value that sustains the adjacent node [3].

$$N_{stability} = \begin{cases} \frac{a}{\sum_{i=1}^a T_i} & (\text{if } a \geq 1) \\ 1 & (\text{if } a = 0) \end{cases} \quad (1)$$

$N_{stability}$: Articulation nodes Stability in Network
 a : Number of Articulation node
 T_i : Number of Adjacent node

2.3 Network Stability Measurement Using $N_{stability}$, Standard Deviation, and Distribution

Network stabilities ($N_{variance}$) are measured by obtaining $N_{stability}$, an improved version of the network stability ($N_{stability}$) measurement of articulation nodes and adjacent trees, the standard deviation, and the distribution results. [3].

$$N_{variance} = \begin{cases} \sqrt{\frac{\sum_{i=1}^a (T_i - \frac{1}{N_{stability}})^2}{a}} & (\text{if } a \geq 1) \\ 0 & (\text{if } a = 0) \end{cases} \quad (2)$$

$N_{variance}$: Variance of Network Stability
 $N_{stability}$: Articulation nodes Stability in Network
 a : Number of Articulation node
 T_i : Number of Adjacent Tree of the Articulation Node

3 Expansion of the Network Stability Measurement Method Based on Articulation Nodes Using Virtual Nodes

3.1 Virtual Node and Virtual Articulation Node

Up until now, we measured the stability of networks with restricted scopes based on the stability of articulation nodes. However, networks are not solely consisted of articulation nodes as non-articulation nodes can become critical nodes of a network.

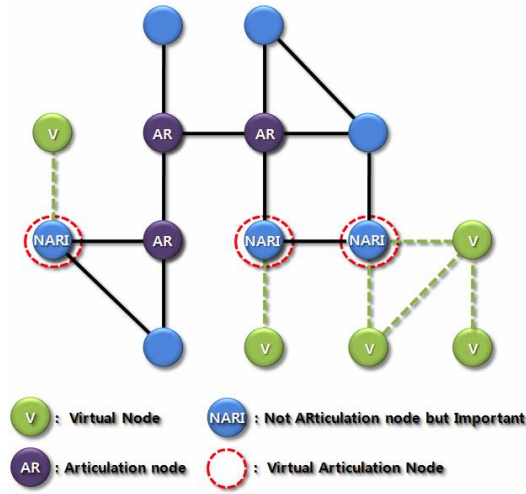


Fig. 1. Addition of Virtual Nodes to NARI Nodes

For example, although nodes that entail special meanings such as servers or routers connected to external networks are not articulation nodes, they have a very high degree of importance. We define these nodes as NARI(Not Articulation Node but Important) nodes.

3.2 Classification of NARI Node Weights

NARI nodes are classified in various ways according to their properties and functions where their degrees of importance vary from one classification method to another.

Table 1. Example of NARI Node Weight Classification

Type	Weight	Number of Virtual Node
External Network	High	9
	Medium	8
	Low	7
Server, Firewall, Mainframe, Supercomputer	High	6
	Medium	5
	Low	4
Host	High	3
	Medium	2
	Low	1

4 Conclusion

By including NARI nodes, which weren't considered in previous measurement methods, and by altering the weight, a more precise network stability level

measurement that reflects the requirements of network administrators is possible. The method that applies NARI nodes can also be used to predict network stability levels when designing networks and for maintenance.

However, we limited the types of NARI nodes that we proposed in this study. Therefore, the diverse range of NARI nodes needs to be taken into account. We will validate the feasibility of the weights of NARI nodes and continue with reliable studies on NARI node weight through specialization.

References

1. Kim, Y., Ahn, S., Chung, J.-w.: Importance of network Articulation node of traffic base analysis techniques. In: Proceedings of the Digital Contents Society, December 2007, pp. 113–117 (2007)
2. Kim, Y.h., Kim, K., Lim, C.S., Ahn, S., Chung, J.-w.: Technique of Analysis of Network's Importance through Analysis on the Relationship between an Articulation Node and its Adjacent Nodes. In: GESTS, pp. 1–8 (January 2008)
3. Kim, Y., Kim, H., Chung, J.-w., Cho, K.-j., Yu, K.-s.: Network Stability Assessment using the Number of Tree Adjacent to an Articulation Node. In: Gervasi, O., Murgante, B., Laganá, A., Tanar, D., Mun, Y., Gavrilova, M.L. (eds.) ICCSA 2008, Part II. LNCS, vol. 5073, pp. 1229–1241. Springer, Heidelberg (2008)
4. Ahn, B., Chung J.-w.: A Study on the Scheme for Gathering Network Management Information using Biconnected Component Computation in internet: Thesis of Master course, Sungkyoungwan Univ. (2001)
5. Horowitz, E., Sahni, S., Anderson-Freed, S.: Fundamentals of Data Structure in C, 2nd edn., pp. 265–292. Silicon Press (August 2007)
6. Aho, A.V., Hopcroft, J.E., Ullman, J.D.: Data Structures and Algorithms. Addison-Wesley, Reading (1983)
7. Cormen, T.H., Leiserson, C.E., Rivest, R.L.: Introduction to Algorithms. McGraw-Hill, New York (1990)
8. Gabow, H.N.: Path-based depth-first search for strong and biconnected components, Tech. Report CU-CS-890-99, revised version, Dept. of Computer Sciencs, University of Colorado at Boulder (2000)
9. Weiss, M.A.: Data Structures and Algorithm Analysis in C++. Addison-Wesley, Reading (1999)

A Channel Management Framework to Construct User Preferred Fast Channel Change Stream in IPTV*

Md. Mamun-Or-Rashid, Dae Sun Kim, and Choong Seon Hong**

Department of Computer Engineering Kyung Hee University
Seocheon, Giheung, Yongin, Gyeonggi, 446-701 Korea
{mamun,dskim,cshong}@khu.ac.kr

Abstract. This paper proposes an effective channel management framework which analyzes the user channel surfing behavior and constructs fast channel change stream consisting of user preferred channels. The usual channel delay is obliterated if the user requested channel is found at the Set-Top-Box (STB).

Keywords: IPTV Channel Change, Channel Management.

1 Introduction

IPTV is one of the most upcoming killer applications among the next generation Internet applications. Due to the bandwidth constraint of access network it is impossible to supply all channels to the user premises (i.e. STB). Therefore channel zapping delay is inevitable if the channel is not found at STB. There are two major strategies "(i) to increase the number of I-frames to reduce decoding delay [1][4] and (ii) to ensure channel availability at STB [2][3]" observed in the contemporary research works to reduce channel zapping delay. Considering usual the notion of human being, it is very evident that user's choice in a locality does not vary rapidly. Therefore, ensuring the availability of user preferred channels at STB certainly reduces channel zapping delay. In this paper we propose a framework to acquire channel surfing behavior and an algorithm to mine user preferred channel sequence to reduce channel zapping delay.

2 Proposed Channel Management Framework

Our basic idea (Fig. 1(a)) is to supply as many channels as possible at the STB based on available bandwidth. To analyze users channel surfing behavior and to prepare the list of preferred channels we need (i) a data collection framework and (ii) channel mining algorithm.

* This research was supported by the MKE under the ITRC support program supervised by the IITA" (IITA-2008-(C1090-0801-0002)).

** Corresponding author.

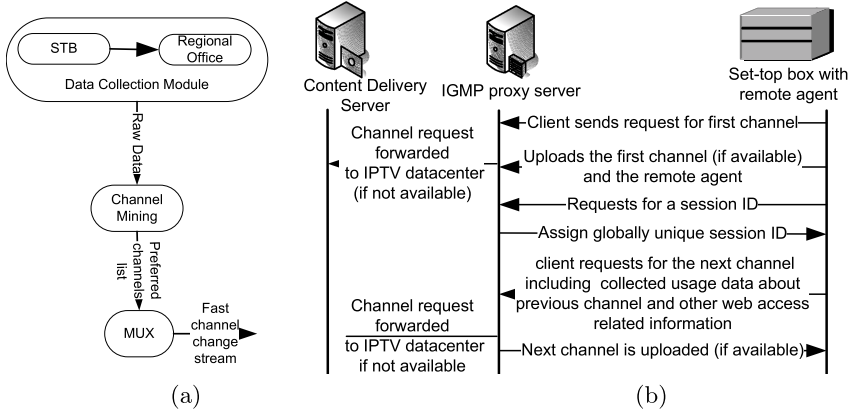


Fig. 1. (a) Overall architecture of the proposed channel management framework
(b) Usage data collection

(i) **Usage Data Collection Framework:** This component is responsible for collecting channel usage data e.g. channel surfing, view time, etc. of the end users. Proposed data collection framework has following two components:

- **STB Agent:** Remote agent resides in the end user's set-top box and supplies load-time, unload time, view-time, etc. Each remote agent is assigned with a unique ID by the usage-data-collector to accumulate the channel usage behavior statistics. As the agent runs at the client's end it must be light and maintain user privacy.
- **Usage-Data-Collector:** Usage-Data-Collector is the module to receive information from the remote agent and directly dump into the database. The operations between STB agent and usage-data-collector are described in Fig. 1(b).

(ii) **Channel Preference Mining Algorithm:** In this section we present utility based channel mining algorithm where utility represents the significance of a particular/group of channels viewed by the end user in terms of viewing length and frequency.

- **Objective value** [$O(ch_i, T_j)$] We define the objective value of ch_i in transaction T_j in the database as a temporal feature. Therefore, it indicates amount of time that a channel is viewed in a particular session.
- **Subjective value** [$S(ch_i, T_j)$] It is defined as the spatial feature and indicates the number of times the channel ch_i is viewed in transaction T_j . In other words it is the viewing frequency of a particular channel for a session.
- **Utility Function** [$U(ch_i, T_j)$] Basically utility function is defined as a function of two real numbers and through any basic operation it produces another positive real number. In our channel mining algorithm we define utility function as the multiplication of subjective and objective value, formally,

$$U(ch_i, T_j) = O(ch_i, T_j) \times S(ch_i, T_j) \quad (1)$$

It is very evident that we can obtain channel preference for a particular transaction in terms of temporal and spatial significance from an end user point of view. Note that the amount of channel to be supplied to the user STB depends on the bandwidth of the access link. Therefore, number of preferred channels to be obtained from the mining algorithm is restricted by the total bandwidth. Let b_{link} be the bandwidth of the access link, so the number of channels to be supplied to the end user can be calculated as:

$$num(ch) = \frac{b_{link}}{b_{ch}} \quad (2)$$

Where $num(ch)$ is the total number of channels that can be supplied to the STB and b_{ch} is the bandwidth required to serve one channel. After obtaining $num(ch)$, our framework executes the following algorithm to determine the list of preferred channels P_{ch} for a particular user.

Channel Mining Algorithm

Input: set of transactions $T_1, T_2, T_3 \dots T_n$ for a particular user
Output: set of preferred channels P_{ch}
var: $P_{ch} = 0, C_{ch} = num(ch), max = 0$
1. **for each transaction** $T_1, T_2, T_3 \dots T_n$
2. $U(ch_i, T_j) = O(ch_i, T_j) \times S(ch_i, T_j)$
3. **end for**
4. **for each transaction** $T_1, T_2, T_3 \dots T_n$
6. $U(ch, T_j) = \sum U(ch_i, T_j) \quad ch_1, ch_2 \dots ch_n \in C_{ch}$
7. **if** $(U(C_{ch}, T_j) \geq max)$
8. $max = U(C_{ch}, T_j); P_{ch} = C_{ch}$
7. **end if**
8. **end for**

The algorithm provides a set of channels having highest utility value. IPTV distribution router multiplexes preferred list of channels obtained from the channel mining algorithm and supplies to the end user's STB. If an end user switches channel within the preferred list i.e. cache hit, the channel can be immediately viewed without any zapping delay.

3 Performance Evaluation

To evaluate the performance of the mining algorithm we have performed simulation in C++. We have considered 30 channels, 15 users and each user channel surfing behavior is analyzed for last 50 transactions in the database. We have generated the channel surfing sequence with a random number and viewing time of each channel is also generated using a random number between 1 to 120. It is to be mentioned that viewing time is measured in 'minute' time unit. Figure 6

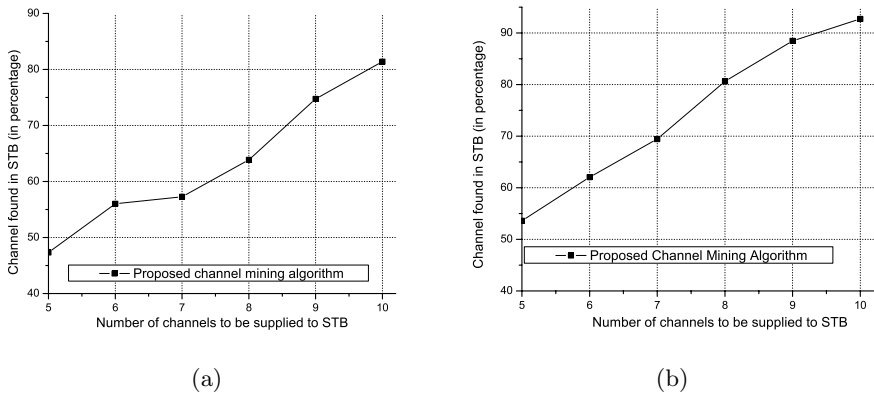


Fig. 2. Channel availability at STB: (a)15 user and (b)10 user

shows channel availability at STB based on our channel mining algorithm. From the result (Fig. 6 (a)) it is evident that channel availability increases if we can supply more channels to STB. Also, channel surfing behavior varies with the amount of user. As Fig. 6(b) shows channel availability with 10 users are more than channel availability with 15 users.

4 Conclusion

Most of the client/end user has a list of favorite channels that he/she watches often and cycles through. Therefore, channel preference mining is of enormous importance in IPTV environment. In this paper we introduce a preliminary idea i.e. utility based channel mining algorithm. In future we would like incorporate statistical measures with the utility mining to obtain higher hit ratio of the user requested channel at the STB.

References

1. Joo, H., Song, H., Lee, D.B., Lee, I.: An effective IPTV channel control considering channel zapping time and network utilization. *IEEE Transaction on Broadcasting* 54(2), 208–216 (2008)
2. Cho, C., Han, I., Jun, Y., Lee, H.: Improvement of channel zapping time in IPTV services using the adjacent groups join-leave method. In: 6th International Conference on Advanced Communication Technology, vol. 2, pp. 971–975 (2004)
3. Jieun, L., Geonbok, L., Seunghak, S., Byngdeok, C.: Advance scheme to reduce IPTV channel zapping time. In: Ata, S., Hong, C.S. (eds.) *APNOMS 2007*. LNCS, vol. 4773, pp. 235–243. Springer, Heidelberg (2007)
4. Jennehag, U., Pettersson, S.: On Synchronization Frames for Channel Switching in a GOP-Based IPTV Environment. In: 5th IEEE Consumer Communications and Networking Conference, CCNC, pp. 638–642 (2008)

A Fair Mobile Payment Protocol

Wei Fan, Huaying Shu, Qiang Yan, and Fang Wang

School of Economics and Management, Beijing University of Posts and
Telecommunications, Beijing 100876, China
{fanwei, shuhuaying, yanqiang, wangfang}@buptinfo.com

Abstract. Based on concurrent signature algorithm, a fair mobile payment protocol is proposed in the paper. The protocol possesses the characters of fairness and non-repudiation. It can protect both of the consumer and merchant's profits. As the electronic check is transferred by publishing a secret key, it guarantees that each participant could get the expected item at the same time. Compared to the traditional process, there are less transaction steps in our scheme. Moreover it can avoid network congestion caused by a traditional TTP and decrease the transaction center's processing pressure much. It could be used for physical goods transaction.

Keywords: Mobile payment protocol, Concurrent signatures, Fairness, Non-repudiation.

1 Introduction

Fairness is a critical property for electronic commerce [1]. It means that at the end of exchange, either both the parties obtain each other's items or none do. Gradually, the research objective is getting weaker, that is aim to gather evidence during protocol execution using which an honest party can prove his case [2].

In general, fair exchange protocols require the continuous availability of an external trusted third party (TTP), a dedicated site which is trusted by both participants [3]. Most of the existing E-payment protocols use an on-line or off-line TTP [4-7] or semi-trusted third party [8-9] to assure the fairness of a transaction. There exist some disadvantages: firstly, the third party is apt to become the bottle-neck of a transaction and lead to network congestion; secondly, TTP decreases the protocol's working efficiency and increases the cost of a transaction; last but not least, it is not very easy to search a third party to serve as the TTP at anytime. If we can design a protocol without TTP, all these problems can be solved. In this paper, by using a kind of concurrent signatures [10], we present a fair mobile payment protocol. It possesses characters of fairness, non-repudiation and security. There is no traditional TTP in this protocol, and it could be used in physical goods transaction.

The rest of this paper is organized as follows. We make a basic introduction about concurrent signature first in section 2, then propose our scheme and analyze its fairness and other characteristics in section 3 and 4 respectively. Finally, the paper's conclusion and future work is given in section 5.

2 Concurrent Signatures

Concurrent signature algorithm is a digital signature scheme comprised of four algorithms. It exploits the ambiguity property enjoyed by ring signatures [11] and designated verifier signatures [12]. Two signers create ambiguous signature to exchange first. Only after an extra piece of information is published (that is keystone), both the ambiguous signatures will bind with their true signers and become valid concurrently. The algorithm detail can be found in the reference [10].

3 The Proposed Protocol

Before the transaction starts, the client Alice should open an electronic check (e-check) operation first. The protocol follows as below:

Step 1. Alice chooses the products she wants to buy after browsing merchant Bob's website, and downloads an order form. Then she chooses a secret key k (as a keystone to hold), and gets $f = H_2(k)$. She will use f to compute the concurrent signature S_A of e-check C , whose par value is equal to the products' price.

$$S_A = (w_A, h_A, f, y_A, y_B, C) \quad (1)$$

In the equations above, $h = H_1((g^t y_B^f \bmod p) \| C)$, $h_A \equiv (h - f) \bmod q$, $w_A \equiv (t - h_A x_A) \bmod q$; t is a random number; H_1 and H_2 are hash functions; x_A , x_B and y_A , y_B are private keys and corresponding public keys of Alice and Bob respectively.

After signing the order form with RSA signature, Alice sends it to Bob together with the e-check C 's concurrent signature S_A .

Step 2. Upon receiving the signed order and e-check from Alice, Bob will validate Alice's identity by the order form using RSA algorithm and examine the e-check's balance information. If the par value is not enough, the transaction terminates and failing information will be sent to Alice.

After validating Alice's identity and account information, Bob will use f to compute concurrent signature S_B of the service guaranty G , and send S_B to Alice.

$$S_B = (w_B, h_B, f, y_B, y_A, G) \quad (2)$$

The equations 2 above could be got similar to equation 1.

Step 3. Upon receiving the signed service guaranty from Bob, Alice will use the averifying algorithm to validate S_B . If she is satisfied with service guaranty, she could send a request to ask Bob to deliver the products.

Step 4. Bob asks deliveryman to send products to Alice.

After the deliveryman arrives at Alice's appointed place with goods ordered, Alice will check the quality of the product. If she is satisfied (or the product quality matches with service guaranty), she will show the signed service guaranty and keystone k to the deliveryman. The deliveryman will use concurrent signature's verifying algorithm to validate Bob's signature on the guaranty. If the guaranty is indeed signed by Bob, the deliveryman will ask Alice to sign for the goods and send keystone k to Bob.

Step 5. Bob validates that Alice has received the goods and will use keystone k to transfer the e-check C to his account.

Upon receiving k , the bank can verify the validity of e-check C . Only if Bob is a legal acceptor of C , the bank will agree the transference. Otherwise, the transfer will fail. At the same time, Bob could inform deliveryman of the successful transfer.

4 Protocol Analysis

The analysis of our scheme is as below.

Fairness. Before Alice publishes k , she can check the product's quality first and compare it with the service guaranty. If she dissatisfies the goods, she could reject publishing the key and accepting products. If she dissatisfies with the products' quality after a period of using or the product is not the same as what the merchant promises, she can also charge against Bob with the service guaranty which is signed by him.

If Alice has validated that the physical goods is the same as the service guaranty describes, but she refuses publishing k , then the deliveryman will not give the goods to her. At worst, if Bob does not ask someone to send the products at all, Alice won't publish k , so Bob can not transfer the check without k .

Non-repudiation. After transaction, Alice can't deny she has received the product, as before she publishes k to Bob, she has signed for goods. Also Bob can't deny he has taken the check, as only when somebody offers the right key to the bank can he transfer the check to his account, and Bob is the only acceptor of the check. If the check has been transferred, it must be done by Bob.

Security. If an attacker intercepts the signed e-check during the transaction, there will be no loss for Alice and Bob. As the keystone is not published to others and the check's acceptor is one and only, nobody can transfer it to one's account without keystone. As the deliveryman just acts as a role that transports goods, and could not transfer the e-check even if he gets the keystone, so the conspiracy problem between deliveryman and client can be avoided.

So this scheme increases requirements for merchants' credit and the quality of products, and it can protect clients' benefit well. Generally speaking, compared with merchants, clients are vulnerable groups in e-commerce. It is better to solve the probable problems before transferring e-check than to solve them after the transaction has finished. In conclusion, the scheme can guarantee transaction's fairness, non-repudiation and security.

5 Conclusion

We have proposed a fair e-payment protocol which is applicable to mobile commerce. Both of the client and merchant's benefits could be protected. The concurrent signature technique applied to our protocol not only simplifies the transaction process, but also make the scheme satisfy security properties, including fairness, non-repudiation and security. Moreover, we get rid of the traditional TTP.

However, in our protocol, whether client publishes the keystone determines whether the electronic check becomes valid. Thus the client has more initiative in the transaction. The future work includes how to obtain absolute fairness as well as efficiency in order to apply the scheme in a more resource-limited environment.

Acknowledgement. This work was supported by the National Natural Science Foundation of China (Grant No. 70671010) and the Key Project of Chinese Ministry of Education (Grant No. 108011).

References

1. Hao, W., He-qing, G.: Fair Payment Protocols for E-Commerce. In: Building the E-Service Society, vol. 146, pp. 227–245. Springer, Boston (2004)
2. Ray, I.: Fair exchange in e-commerce. *ACM SIGecom Exchange* 3(2), 9–17 (2002)
3. Vogt, H., Gartner, F. C., Pagnia, H.: Supporting Fair Exchange in Mobile Environments. *Mobile Networks and Applications* 8(2), 127–136 (2003)
4. Asokan, N., Shoup, V., Waidner, M.: Optimistic Fair Exchange of Digital Signatures. *IEEE Journal on Selected Areas in Communication* 18(4), 593–610 (2000)
5. Park, J., Chong, E., Siegel, H.: Constructing Fair-Exchange Protocols for E-Commerce via Distributed Computation of RSA Signatures. In: *Proceedings of PODC 2003*, pp. 172–181. ACM Press, New York (2003)
6. Zhou, J., Gollmann, D.: An Efficient Non-repudiation Protocol. In: *1997 IEEE Computer Security Foundations Workshop*, pp. 126–132. IEEE Computer Society Press, Los Alamitos (1997)
7. Bao, F., Deng, R., Nguyen, K.Q., Varadharajan, V.: Multi-party fair exchange with an off-line trusted neutral party. In: *10th International Workshop on Database & Expert Systems Applications*, pp. 858–862 (1999)
8. Franklin, M.K., Reiter, M.K.: Fair exchange with a semi-trusted third party. In: *4th ACM conference on Computer and Communications Security*, pp. 1–5. ACM Press, New York (1997)
9. Zheng, D., Chen, K., You, J.: Multiparty Authentication Services and Key Agreement Protocols with Semi-Trusted Third Party. *Journal of Computer Science and Technology* 17(6), 749–756 (2002)
10. Chen, L., Kudla, C., Paterson, K.G.: Concurrent signature. In: Cachin, C., Camenisch, J.L. (eds.) *EUROCRYPT 2004*. LNCS, vol. 3027, pp. 287–305. Springer, Heidelberg (2004)
11. Rivest, R., Shamir, A., Tauman, Y.: How to leak a secret. In: Boyd, C. (ed.) *ASIACRYPT 2001*. LNCS, vol. 2248, pp. 552–565. Springer, Heidelberg (2001)
12. Jakobsson, M., Sako, K., Impagliazzo, R.: Designated verifier proofs and their applications. In: Maurer, U.M. (ed.) *EUROCRYPT 1996*. LNCS, vol. 1070, pp. 143–154. Springer, Heidelberg (1996)

Statistical Analysis of Slow Portsweep

Noriaki Yoshiura

Department of Information and Computer Science, Saitama University
255, Shimo-ookubo, Sakura-ku, Saitama City, Saitama Prefecture, Japan
Tel.:+81-48-858-3498; Fax:+81-48-858-3498
yoshiura@fmx.ics.saitama-u.ac.jp

Abstract. This paper reports statistical analysis of portsweep, especially, slow portsweep. Since we need long span traffic data in order to analyze slow portsweep, we collected traffic data during twenty-five months by one PC having sixteen IP addresses on a network interface. The report of this paper seems useful for network attack detection or protection, especially for long span network attacks.

1 Introduction

Computer network attack becomes one of the most important social problems as Internet spreads widely all over the world. Portsweep is one of the ways of finding targets. If an attacker finds several hosts in which a specific listening port is available, it sends to those hosts anomaly data. Intrusion Detection System (IDS) or Intrusion Protection System (IPS) detects and protects portsweeps and anomaly traffics [4,5,8,9].

There are several kinds of portsweeps. The simplest portsweep is to scan multiple hosts in IP address order from an attack host within a short time, however, this portsweep is easy for IDS or IPS to detect. In order to avoid detection, attackers make improvements of portsweeps. One of them is that the attackers scan multiple hosts in a long time span. It is called slow portsweep, which is difficult for IDS to detect. This paper analyzes portsweep tendency from traffic data, which are collected by one PC having sixteen IP address during twenty-five months (from 1st March 2005 to 31st March 2007). Since the data is a long spam data, this paper can focus slow portsweep.

Traffic data analysis has been done in Emergency Response Team Coordinator Center (CERT) [10] and many other organizations, and several organizations such as Japanese Police Agency observes Internet traffic at many points, but do not focus tendency of slow portsweep. There are several researches of optimization of firewall rules by using traffic data[1,2,3]. In order to evaluate such optimization, tendency of portsweep is necessary.

There are a few researches of slow portsweep. In [7], the interval between portsweeps is supposed to be six hundred seconds. However, we cannot known that this interval setting is suitable. In this paper, we analyze twenty-five month traffic data in order to examine the suitable interval setting.

The way of data collection in this paper is cheap; we use only one usual PC. Generally, traffic data collection needs special equipments if we need detailed information of traffic. We can also use TCPDUMP on PC in order to collect traffic data. However, such equipments or methods are expensive or traffic data are too large to analysis. Cisco Netflow can reduce the traffic data, however, we would need some kind of Cisco equipments. Honeypot is one of tools of detecting anomaly traffic, however, it is difficult to use and it is not the aim of honeypot to obtain all access data from attackers.

One of the problem of traffic analysis is size of data. Monitoring even a small network with a few hundred hosts can generate many gigabytes of TCPDUMP data per day. Several methods are proposed [6]. Traffic data dealt in this paper include only access packet information but not session information. It follows that analysis of twenty-five month traffic data is not a hard work.

2 Data Collection

In order to collect traffic data, we prepare one FreeBSD PC having sixteen IP addresses (XXX.XXX.XXX.16~ XXX.XXX.XXX.31) on a network interface. By ipfw (firewall tools of FreeBSD), we collect all traffic data access to these IP addresses. The PC does not response any access, and thus we collect only access traffic data. The collection duration is twenty-five months from 1st March 2005 to 31st March 2007. Traffic data which are collected by the PC include source IP address, source port number, destination IP address, destination port number, date and a kind of traffic such as TCP/UDP/ICMP. The size of the traffic data is 60Mbyte.

The basic data is as follows; the total number of access is 628,582, the total number of access IP source address is 58,737 and the total numbers of TCP, UDP, and ICMP port access are 455,002, 154,846 and 18,734 respectively. From 44,683 IP addresses in 58,737, the number of portsweeps is less than ten and many portsweeps from one IP address seldom happen. Moreover, 23,730 IP addresses, which is about half of all IP addresses, do only one portsweep and the total number of portsweeps from 23,730 IP addresses occupies only 3.8% of all portsweeps.

3 Slow Portsweep

This section analyzes traffic data from the viewpoint of slow portsweep. From traffic data, the intervals of portsweeps from 35,007 IP addresses in 58,737 are less than one second, however, we find that there are several slow portsweeps. Long interval portsweeps of TCP are less than those of UDP even if TCP portsweeps are much more than UDP portsweeps. Since we use sixteen IP addresses to collect traffic data, the attackers who have strong intentions of intrusion is supposed to make more than or equal to sixteen time portsweeps. In the TCP case, the intervals of more than or equal to sixteen time portsweeps are less than 21,600 seconds. It follows that after we detect one portsweep of a certain IP

address, we have to keep this information within 21,600 seconds to detect slow portsweep. In fact, the number of portsweeps whose interval is less than 21,600 seconds is 26,572, which is 99% of all of the TCP portsweeps which make more than one portsweeps. In the UDP case, there are several slow portsweeps. The number of portsweeps whose interval is less than 21,600 seconds is 5,636, which is 75% of all of the UDP portsweeps which make more than one portsweeps. Therefore, after we detect one portsweep of a certain IP address, we have to keep this information to detect slow portsweep in UDP traffic longer than in TCP traffic.

4 IP Addresses Making Portsweeps to All Prepared IP Addresses

We prepare sixteen IP addresses in order to collect traffic data. This section focuses IP address which alone makes portsweeps to all prepared IP addresses. Table 1 shows the number of IP addresses which alone makes portsweeps to all prepared IP addresses for portsweep interval. As this table shows, with respect to portsweeps to all prepared IP addresses, the number of source IP addresses is 10,243 and the number of portsweeps is 457,781, which is 73% of all portsweeps.

Next, we check the data of TCP, UDP and ICMP respectively. The number of portsweeps and the number of IP address in TCP are much more than those in UDP and in ICMP. For all portsweeps to all prepared IP address, the number of portsweeps in TCP occupies 78% and the number of IP address in TCP occupies 82%. From the viewpoint of portsweep interval, there is no portsweep whose interval is more than 7,200 second in TCP, however, there are portsweeps whose interval is more than 86,400 in UDP.

4.1 Identifying Several IP Addresses by Using Network Mask

Portsweep from one IP address to all prepared IP address is easier for IDS to detect than that from several IP addresses. Thus, one of ways of avoiding IDS detection is to distribute source IP addresses of portsweep, that is to say, to make portsweeps to all prepared IP addresses from several IP addresses.

To investigate such a portsweep, we identify IP addresses if they are the same as to first, second and third octets of IP address, that is to say, we use 24 bit length network mask in order to identify several IP addresses as one attack source IP network address. By using network mask, we obtain the following information; the number of source IP network addresses is 45,272. The number of source IP network addresses making portsweeps to all prepared IP addresses is 9,722 in 45,272. The number of portsweeps from 9,722 source IP network addresses is 484,388. The number of portsweeps except the number of portsweeps from one IP address which alone makes portsweeps to all prepared IP addresses is 26,607. The number of IP network addresses which make portsweeps to all prepared IP addresses and whose each IP address does not make portsweeps to all prepared IP addresses is 191. The number of the portsweep source IP addresses which belong to 191 IP network addresses is 2,466.

Table 1. The data of portsweeps to all prepared IP addresses

interval	portsweep times	IP address	TCP portsweep times	IP address	UDP portsweep times	IP address	ICMP portsweep times	IP address
0	457781	10243	352566	8398	94927	1599	9777	255
1	70119	612	10688	172	59923	440	156	5
60	58886	445	340	15	58585	431	94	2
600	42062	354	56	3	41980	351	94	2
3600	19765	191	24	1	19647	188	94	2
7200	11587	122	0	0	11493	120	94	2
7300	11534	120	0	0	11440	118	94	2
7400	10972	118	0	0	10878	116	94	2
7500	10673	115	0	0	10579	113	94	2
7600	10620	114	0	0	10526	112	94	2
7700	10552	113	0	0	10458	111	94	2
7800	5162	77	0	0	5068	75	94	2
7900	4195	63	0	0	4101	61	94	2
10800	1586	29	0	0	1492	27	94	2
21600	727	16	0	0	633	14	94	2
43200	261	8	0	0	167	6	94	2
86400	110	3	0	0	74	2	36	1
172800	0	0	0	0	0	0	0	0

5 Conclusion

This paper reported statistical analysis of portsweep, especially slow portsweep, by analyzing twenty-five month traffic data. From this result, we can estimate a suitable interval of slow portsweep.

References

1. Acharya, S., Wang, J., Ge, Z., Znati, T.F., Greenberg, A.: Traffic-Aware Firewall Optimization Strategies. In: The proceedings of IEEE International Conference on Communications, pp. 2225–2230 (2006)
2. Acharya, S., Abliz, M., Mills, B., Znati, T.F., Wang, J., Ge, Z., Greenberg, A.: OPTWALL: A Hierarchical Traffic-Aware Firewall. In: Proceedings of the Network and Distributed System Symposium (2007)
3. Hamed, H., Al-Shaer, E.: Dynamic Rule-ordering Optimization for High-speed Firewall Filtering. In: Proceedings of the 2006 ACM Symposium on Information, computer and communications security, pp. 332–342 (2006)
4. Kou, G., Peng, Y., Shi, Y., Chen, Z.: Network Intrusion Detection by Multi-group Mathematical Programming based Classifier. In: Proceedings of Sixth IEEE International Conference on Data Mining, pp. 803–807 (2006)
5. Denning, D.: An intrusion detection model. IEEE Transactions of Software Engineering SE-13(2), 222–232 (1987)
6. McHugh, J.: Sets, Bags and Rock and Roll - Analysing Large Data Sets of Network Data. In: WAIM 2004. LNCS, vol. 3129, pp. 407–422. Springer, Heidelberg (2004)
7. Malmedal, B.: Using Netflows for slow portscan detection, Master's thesis, Gjøvik University Collage (2005)
8. Snort - An open source network intrusion detection system, <http://www.snort.org/>
9. Cisco Intrusion Detection, <http://www.cisco.com/warp/public/cc/pd/sqsw/sqidsz/index.shtml>
10. Emergency Response Team Coordinator Center, <http://www.cert.org/>

Empirical Analysis of Application-Level Traffic Classification Using Supervised Machine Learning

Byungchul Park¹, Young J. Won¹, Mi-Jung Choi¹,
Myung-Sup Kim², and James W. Hong¹

¹Dept. of Computer Science and Engineering, POSTECH, Pohang, Korea
{fates, yjwon, mjchoi, jwkhong}@postech.ac.kr

²Dept. of Computer and Information Science, Korea University, Jochiwon, Korea
tmskin@korea.ac.kr

Abstract. Accurate application traffic classification and identification are important for network monitoring and analysis. The accuracy of traditional Internet application traffic classification approaches is rapidly decreasing due to the diversity of today's Internet application traffic, such as ephemeral port allocation, proprietary protocol, and traffic encryption. This paper presents an empirical evaluation of application-level traffic classification using supervised machine learning techniques. Our results indicate that we cannot achieve high accuracy with a simple feature set. Even if a simple feature set shows good performance in application category-level classification, more sophisticated feature selection methods and other techniques are necessary for performance enhancement.

Keywords: Internet application traffic identification, traffic measurement and analysis, machine learning, supervised algorithm.

1 Introduction

Accurate traffic classification based on application is an important step toward providing an informative snapshot of networks with respect to various network operation and management aspects such as traffic shaping, monitoring, capacity planning, charging, QoS management, and so on.

Traditional Internet traffic classification approaches rely on well-known port mapping and application signature mapping [1, 2]. However, the newer generation of P2P applications is incorporated with various sneaking-by strategies such as ephemeral port allocation and relay node to avoid detection and filtering. Moreover, emerging network applications such as Skype are eluding detection by packet payload encryption or plain-text ciphers [3]. Privacy legislation related to payload inspection is also a problem in this approach.

To overcome the limitation of port-based classification and signature-based classification, machine learning techniques that use statistical information of the transport layer have been introduced recently [4, 5, 6, 7]. These classification methods depend on the fact that different applications have different communication patterns or behaviors.

In this paper, we narrow the classification level down to the specific application level, in contrast to previous research, and we explore the possibility of supervised machine learning technique for enabling traffic classification by empirical evidence. In addition, we point out the possible problem of traffic classification using machine learning techniques.

The remainder of this paper is organized as follows. Section 2 presents an overview related work with problem domain. In Section 3, selected supervised learning algorithms, feature sets, and data used in our work are covered. Finally, we conclude our work with some suggestions of possible future work in Section 4.

2 Problem Domain

Many researchers have attempted to use machine learning algorithms for traffic classification [4-10]. Erman *et al.* [4] used three different unsupervised algorithms (K-Means, DBSCAN, Autoclass) and other researchers used various supervised learning algorithms. Each researcher observed that these algorithms show good performance for traffic classification in terms of classification accuracy (maximum 92~93%).

The classification target of these works is somewhat different from our research goal. Although the common aspect of these studies may be that machine learning techniques were used in traffic classification, the depth of the classification level in them is different. Prior works classified network traffic based on the application type (Bulk, Database, Mail, Web, P2P, etc) or application protocol (DNS, RTP, HTTP, POP3, etc). While our objective is to target more sophisticated application-level classification using supervised machine learning algorithms, previous works relied on known and limited means of layer-3 or above protocol formats.

3 Machine Learning Algorithm and Feature Set

To verify the performance of supervised learning in application traffic classification, we selected four different algorithms: J48, REPTree, Multi-layer Perceptron (MLP), and BayesNet. The first three algorithms showed very high accuracy in previous works. MLP is one of the supervised learning algorithms that showed good performance in other data mining applications.

We selected features that were commonly used in other traffic classification works: 1) Source IP address, 2) Destination IP address, 3) Source port number, 4) Destination port number, 5) Total bytes transferred, 6) Connection duration, 7) Packet size (min/max/mean/standard deviation), 8) Packet inter-arrival time (min/max/mean/standard deviation). The target application traffic was aggregated to a flow that is a collection of packets sharing identical 5-tuple (source IP, destination IP, source port, destination port, and protocol) information. After aggregating traffic data into the flow, the features were extracted from the flow.

3.1 Target Applications and Data Collection

We selected 14 popular applications (Table 1). As described in Section 2, our traffic classification targeted more sophisticated application-level classification and not the

application-category level; therefore, several applications belonging to the same category were selected as target applications.

The ultimate goal of traffic classification research is to classify the backbone network traffic; therefore, we collected testing data form POSTECH’s backbone network.

Table 1. Traffic summary of training and testing data

Type	Application	Training data		Testing data	
	Name	# of flow	Size (MB)	# of flow	Size (MB)
Web	Clubbox	231	165	165	102
Storage	ParanDisk	223	278	148	145
	TotoDisk	526	367	247	192
	EnDisk	246	447	550	384
Web	HTTP	463	27	2045	15
App.	HanGame	998	24	47	7
	SayClub	2870	10	85	3
FTP	AlFTP	518	1,011	231	342
P2P	BitTorrent	11,000	708	4,000	256
	Fileguri	571	506	408	407
	Soribada	8803	339	5960	273
	Gample	10,943	990	547	4,032
Game	Starcraft	15	0	6	0
Etc.	MS Remote Desktop	86	28	34	12
Total		37,493	4900	14,473	6170

Table 1 shows the traffic trace summary of each application. In the case of Starcraft, very small amount of traffic data was generated over duration of 30 minutes; consequently, the total amount of traffic was less than 1 MB.

4 Conclusion

From the experiment, we made the following observations. (1) Even if we narrow the level of classification to that of a specific application level, the overall accuracy of the classification is still compatible (80~90%) with the accuracy of the application category-level of classification that was conducted in previous research. (2) However, precision and recall of each class fluctuated from 0 to 1 depending on the application.

From the results obtained by measuring precision and recall, we could conclude that the machine learning algorithms and the feature set that we used are not efficient for application-level traffic classification, even if these algorithms and the feature set

performed very well in classifying application categories due to the difference in the level of classification. Based on the above discussions, we concluded that a feature selection algorithm and a modified machine learning algorithm are essential to overcome these hurdles. There has been much research on feature selection [11, 12], and finding an effective feature selection for traffic classification will help enhance the accuracy of classifiers. For future work, we plan to find the best possible feature set and modify machine learning algorithms so that they are suitable for application-level traffic classification.

Acknowledgments. This work was partly supported by the IT R&D program of MKE/ITA [2008-F-016-01, CASFI] and the EECE division at POSTECH under the BK21 program of MEST, Korea.

References

1. Sen, S., Spatscheck, O., Wang, D.: Accurate, Scalable In-Network Identification of P2P Traffic Using Application Signatures. In: WWW 2004 Conference (2004)
2. Karagiannis, T., Broido, A., Faloutsos, M., Claffy, K.C.: Transport layer identification of p2p traffic. In: Internet Measurement Conference (IMC) (2004)
3. Park, B.-C., Won, Y.J., Kim, M.-S., Hong, J.W.-K.: Towards Automated Application Signature Generation for Traffic Identification. In: Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS 2008), Salvador, Brazil, April 2008, pp. 160–167 (2008)
4. Erman, J., Arlitt, M., Mahanti, A.: Traffic Classification Using Clustering Algorithms. In: SIGCOMM 2006 Workshops, Pisa, Italy, September 2006, pp. 281–286 (2006)
5. Zander, S., Nguyen, T., Armitage, G.: Automated Traffic Classification and Application Identification using Machine Learning. In: Proceedings of the IEEE Conference on Local Computer Networks, Sydney, Australia, November 2005, pp. 250–257 (2005)
6. Thuy, T., Nguyen, T., Armitage, G.: Training on multiple Sub-Flows to optimize the use of Machine Learning classifiers in real-world IP Networks. In: IEEE Conference on Local Computer Networks, Tampa, Florida, USA, November 2006, pp. 369–376 (2006)
7. Park, J., Tyan, H.-R., Kuo, C.-C.J.: GA-Based Internet Traffic Classification Technique for QoS Provisioning. In: International Conference on Intelligent Information Hiding and Multimedia, Pasadena, California, USA, December 2006, pp. 251–254 (2006)
8. Moore, A.W., Zuev, D.: Internet Traffic Classification Using Bayesian Analysis Techniques. In: SIGMETRICS 2005, Banff, Alberta, Canada, June 2005, pp. 50–60 (2005)
9. Erman, J., Mahanti, A., Arlitt, M.: Internet Traffic Identification using Machine Learning. In: IEEE Global Telecommunications Conference, California, USA, November–December 2006, pp. 1–6 (2006)
10. Williams, N., Zander, S., Armitage, G.: A Preliminary Performance Comparison of Five Machine Learning Algorithms for Practical IP Traffic Flow Classification. In: SIGCOMM Computer Communication Review, October 2006, pp. 7–15 (2006)
11. Battiti, R.: Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on Neural Networks* 5(4) (July 1994)
12. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. *Journal of Machine Learning Research* (2003)

Virtualization-Based Operation Support Systems: Performance Evaluation and Systems Design

Yujiro Mochizuki, Kazuhiko Higashi, Kumiko Goto, and Minoru Kato

Open Source Software Department, NTT Comware Corporation,
1-6 Nakase, Mihama, Chiba-shi, Chiba, 261-0023, Japan
{mochi.yu, kazuhiko.higashi, gotou.kumiko,
minoru.kato}@nttcom.co.jp

Abstract. Recent advances in virtualization technologies have allowed telecommunications carriers to apply virtualization-based operation support systems. By separating logical view of computing resources from physical view, virtualization is expected to help consolidate/simplify existing complex telecommunication systems, improve utilization rate and reduce Total Cost of Ownership. While the benefits of virtualization are compelling, potential drawbacks also must be identified before actual deployment. Without proper evaluation, virtualization cannot benefit operation support systems. From the viewpoint of operation support systems, this paper (1) analyses performance issues (CPU, network throughput and Disk I/O) associated with Xen virtualization. Given the current technology level and development progress, this paper (2) identifies applicable areas for virtualization-based operation support systems and addresses suggested system architecture.

Keywords: Virtualization, Xen, Virtual Machine Monitor, Hypervisor, Operation Support Systems, Server Consolidation.

1 Introduction

The last several years have shown a worldwide rise in attention and actual use of virtualization technology. Telecommunications systems are highly distributed and complex with relatively stagnant (20-30% CPU usage) or over-powered servers. Under such situation virtualization technology appears to be reasonably well defined and satisfactory solution.

While the benefits of virtualization are compelling, potential drawbacks of virtualization also must be identified before actual deployment. Virtualization is not an almighty solution. In particular, virtualization is achieved by adding a virtualization layer for resource isolation and control, which allows different operating systems to properly coexist on the same physical hardware. At the same time, however, this layer also incurs performance overheads. Without right systems design, virtualization cannot improve efficiency of the system.

From the viewpoint of operation support systems, this paper focuses on performance issues (CPU, network throughput and Disk I/O) and identifies areas where virtualization-based operation support systems are effectively applied.

2 Related Works and Problem Definition

Several papers and articles have discussed controversial issues about virtualization[1] [2] [3]. In particular, the following papers raise performance issues [4] [5] [6]. Although the related works above clarify fundamental issues associated with virtualization, more discussions are necessary from the viewpoint of telecommunications systems. In this paper we (1) identify potential drawbacks and appropriate system architectures, (2) define the performance evaluation model under the current technology level and development progress and (3) renew quantitative data and analyses in previous studies.

3 Performance Evaluation

Operation support systems usually execute a mix of CPU and I/O (network and disk) intensive applications. In order to identify performance overheads of virtualization, we measured (1) CPU scalability, (2) Network throughput, and (3) Disk I/O performance.

3.1 CPU Scalability

We measured performance of a single guest OS (Linux, 2Gbyte of memory), running on top of the Xen hypervisor up to 8 CPU cores. Figure 1 shows the CPU scalability, comparing native Linux, full virtualization and para virtualization. Unlike previous researches [7], this result indicates that (1) full virtualization can perform like para virtualization and (2) CPU scalability is high in both full virtualization and para virtualization environments. A slowdown over native Linux is ranging from 3-6%.

This phenomenon results from hardware-assisted virtualization, such as Intel Virtualization Technology, which allows virtual machines to directly access CPU and avoid emulation overheads via hypervisor.

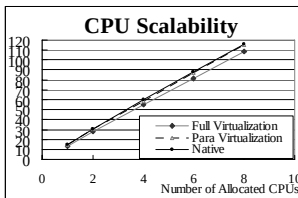


Fig. 1. CPU Scalability

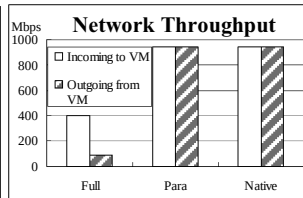


Fig. 2. Network Throughput

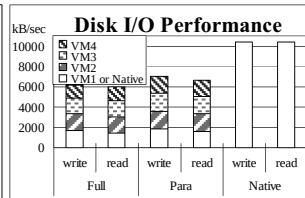


Fig. 3. Disk I/O Performance

3.2 Network Throughput

Figure 2 shows the Netperf results for network throughput of sending and receiving data. Para virtualization and full virtualization deliver almost 100% of native Linux (940Mbps) and 10-40% (91-405Mbps) respectively. After changing the backbone network from 1Gbps to 100Mbps, we found the similar results (full virtualization delivered only 10-30Mbps). We also found that each CPU (core) usage is almost

100% during the test period. In order to avoid this situation, each virtual machine (including domain0) should exclusively use at least one CPU core.

From the viewpoint of stability, para virtualization maintains a stable network throughput during the evaluation period. On the other hand, network throughput of full virtualization is highly unstable. In particular, incoming throughput to virtual machine drastically fluctuates from 20Mbps to 400Mbps. Summarizing the results above, network-intensive applications should avoid full virtualization.

3.3 Disk I/O

As shown in Figure 3, neither full virtualization nor para virtualization can perform well in disk I/O (40% decrease in Disk I/O performance). All of the Disk I/O transactions must be managed in Domain0, a special privileged domain to manage virtualization environment, so severe performance deterioration occurs due to Xen virtualization technology itself. In addition, unlike CPU performance, Disk I/O has no hardware-assisted virtualization functions. Thus, performance deterioration cannot be avoided in both full virtualization and para virtualization.

4 Applicable Areas and Suggested System Architecture

Table 1 summarizes the applicable areas of virtualization technology for operation support systems. Compared with native Linux, virtualization incurs significant overheads in Disk I/O (40% decrease). Given the current technology level and development progress, virtualization technology is not effective in DB consolidation of operation support systems. On the other hand, CPU-intensive applications are suitable for virtualization. In terms of network intensive applications, para virtualization can be applied.

Table 1. Applicable Areas of Virtualization Technology (Xen) in Operation Support Systems

	Full Virtualization	Para Virtualization	Notes
CPU intensive	Recommended	Recommended	Almost native
Network intensive	Not recommended	Can be applied	Para: Stable, Full: Unstable
DB intensive	Not recommended	Not recommended	40% performance decrease

On the basis of the performance evaluation, Figure 4 illustrates the suggested system architecture for virtualization-based operation support systems. Note that we assume 8-core servers (Quad-Cores, 2 Sockets) for implementation according to our field study and actual user requirements of telecommunications carriers.

First, in order to avoid negative performance effects from other virtual machines, each virtual machine should occupy at least one CPU core. In particular, we suggest that Domain0 exclusively use 2 CPU cores to manage virtualization environment.

Second, each virtual machine should be stored in other physical storage. As we discussed in 4.3, Disk I/O is a critical bottleneck of Xen. Separating each virtual machine from others, performance deterioration can be minimized.

Third, 1Gbps Ethernet is recommended. Each virtual machine shares the total throughput, so sufficient network capacity is necessary to operate applications. Given the instability and deterioration of network performance, we especially recommend that 1Gbps Ethernet be necessary in full virtualization environment.

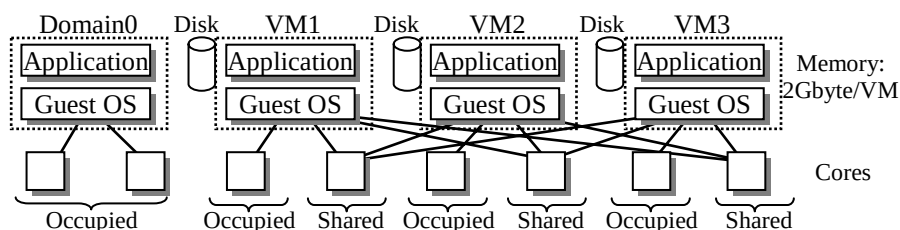


Fig. 4. Suggested System Architecture (Network: 1Gbps)

5 Conclusion

Evolution of virtualization technology has allowed telecommunications carriers to apply virtualization-based operation support systems. While the benefits of virtualization are compelling, potential drawbacks of virtualization also exist. In this paper we focused on performance issues arising from virtualization technology. Our study shows that at present CPU scalability is well developed in Xen. However, I/O virtualization should improve performance for telecommunication systems. Compared with native Linux, virtualization-based operation support systems show 40% and 10-40% performance decrease in disk I/O and network throughput (full virtualization) respectively. Currently, virtualization technology can be applied to CPU-intensive or low traffic areas, such as Web/AP servers in service order processing system and network management in intelligent networks.

Performance improvement remains a future work and challenging issue. However, telecommunications carriers can simplify IT infrastructure, improve utilization rate and reduce operational costs by virtualization technology.

References

1. Crosby, S., Brown, D.: The Virtualization Reality. *ACM Queue* 4(10), 34–41 (2006)
2. Fong, L., Steinder, M.: Duality of Virtualization: Simplification and Complexity. *ACM SIGOPS Operating Systems Review* 42(1), 96–97 (2008)
3. Drepper, U.: The Cost of Virtualization. *Queue* 6(1), 28–35 (2008)
4. Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I., Warfield, A.: Xen and the Art of Virtualization. In: *Proceedings of the nineteenth ACM symposium on Operating Systems Principles*, pp. 164–177 (2003)
5. Clark, B., Deshane, T., Dow, E., Evanchik, S., Finlayson, M., Herne, J., Matthews, J.N.: Xen and the Art of Repeated Research. In: *Proceedings of the USENIX 2004 Annual Technical Conference, FREENIX*, pp. 135–144 (2004)

6. Menon, A., Santos, J., Turner, Y., Janakiraman, G., Zwaenepoel, W.: Diagnosing Performance Overheads in the Xen Virtual Machine Environment. In: Proceedings of the First ACM/USENIX International Conference on Virtual Execution Environments, pp. 13–23 (2005)
7. Theurer, A., Rister, K., Krieger, O., Harper, R., Dobbelsstein, S.: Virtual Scalability: Charting the Performance of Linux in a Virtual World. In: Proceedings of the Linux Symposium, pp. 393–402 (2006)

A Management Framework for IMS Using Service Managed Objects^{*}

Muhammad Shoaib Siddiqui, Syed Obaid Amin, and Choong Seon Hong^{**}

Kyung Hee University, Korea
{shoaib, obaid}@networking.khu.ac.kr, cshong@khu.ac.kr

Abstract. In this paper we identify the needs and critical issues of management and monitoring in IP Multimedia Subsystem (IMS). As an IMS provides a layered architecture converging heterogeneous networks; it requires network management as well as service management. We provide a framework for managing an IMS system by utilizing network management server, session management and service managed objects (SMOs). SMOs are similar to managed objects (MO) except they are used for managing the provided services, error detection and fault recovery in services.

Keywords: Service management, Service managed objects, IMS, SDP.

1 Introduction

IP multimedia subsystem (IMS) is the future for all IP next-generation converged networks with potential of enabling service providers to create and provide value added services to users on heterogeneous networks. IMS was defined by 3GPP [1] as an standard architecture which provides a horizontal, cross-functional layer of intelligence on top of IP, enabling the creation, control and execution of new and rich user-to-user services (video streaming), user-to-server offerings (IPTV) and multi-user media services (game-playing on the move and at home via PC). To enable this, IMS architecture must be made compatible with existing service delivery environment such as Service Delivery Platform (SDP). A service delivery platform helps to standardize all the service interfaces for a provider, creating a horizontal platform from which they can provision, control and bill for all the value-added services they provide. A typical SDP+IMS [2] solution is depicted in figure 1.

In this paper, we have provided a study of management and control of services provided using IP Multimedia Subsystem Architecture. In section 2, we discuss some related work to management of IMS and SDP. In section 3, we discuss the management of IMS and SDP. Section 4 discusses the new concept for the management of IMS and SDP, which we have named Service Managed Objects (SMO). Finally, we conclude our work.

^{*} This research was supported by the MKE under the ITRC support program supervised by the IITA" (IITA-2008-(C1090-0801-0002)"

^{**} Corresponding author.

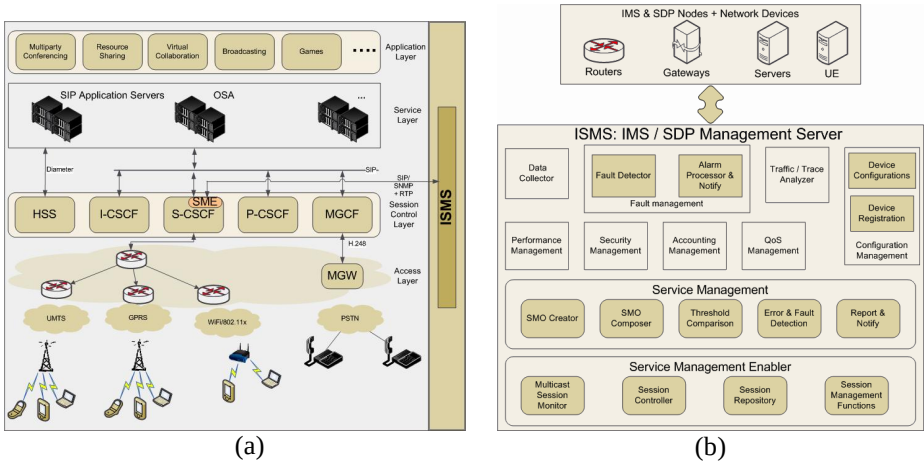


Fig. 1. (a). Simplified view of IMS/ SDP architecture with ISMS as a vertical layer (b). IMS/ SDP Management Server component diagram

2 Related Works

Standards organizations such as ETSI/TISPAN, 3GPP, 3GPP2, OMA, Parlay Group, Java Community Process (JCP) and Internet Engineering Task Force (IETF) are all cooperating to deliver mature open and industry-standard architectural specifications, which spell out what protocols and APIs are needed by CSPs and application developers to facilitate the deployment of IMS and SDP systems. Many companies have introduced their solution of SDP provisioning through IMS, which include IBM, Ericsson, BEA, HP, Motorola and etc.

The Fraunhofer FOKUS Institute opened in July 2004, the "Open IMS Playground" [3]. The FOKUS Open IMS Playground is an open test environment, all major IMS components (especially the FOKUS Open IMS Core). The components come from their own development and from leading industry partners. Open IMS has the IMS Management architecture, which provides means for monitoring and controlling all vital IMS core network components. Open IMS use active and passive traffic generation to control and monitor the IMS environment.

All the available IMS solutions use conventional traffic analysis and performance parameters to manage the IMS and SDP environment. After analyzing the available systems for SDP using IMS, we came to a conclusion that a comprehensive management framework is required which can ease up the management process in all aspect starting from service delivery to network performance and that more emphasis should be given to the management and control issues of IMS and SDP with three major aspects, which are, network management, service management and session management.

3 Management in SDP/IMS

Management Server (ISMS) is a fundamental part of OSS as shown in figure 1-a. It supervises the vital states of IMS/SDP components. IMS/SDP process and traffic are

constantly monitored, both actively and passively. The captured traffic is collected, correlated and analyzed and performance information of session control layer and quality of delivered services is diagnosed. Management server also gathers different performance, security, fault and configuration parameters from all over the network to maintain a healthy state of the network. For this it uses Agents at each managed nodes to deliver it the required information.

4 Services Managed Objects

Service Level Managed Object is a concept to define the service parameters which identify the performance, configuration, faults, accounting and security parameters. As the service is delivered by the service provider with help of different nodes within the system, service parameters are gathered throughout the network. A SMO may consists of managed object from the Network Element level parameters gathered from IMS or SDP nodes or an SMO may be consisted of different SMOs. A high level SMO, which is composed of low level SMOs and final at the lowest level there are some Managed Objects.

High level SMOs are made of generic terminologies which are understandable to service provider, operator and as well as the customer. As the SMOs of lower level are defined, they become more and more technology specific. Given is an example of an SMO defined for a session disruption in IPTV service. The higher level SMO is defined and then more detailed SMOs are defined (one for each level).

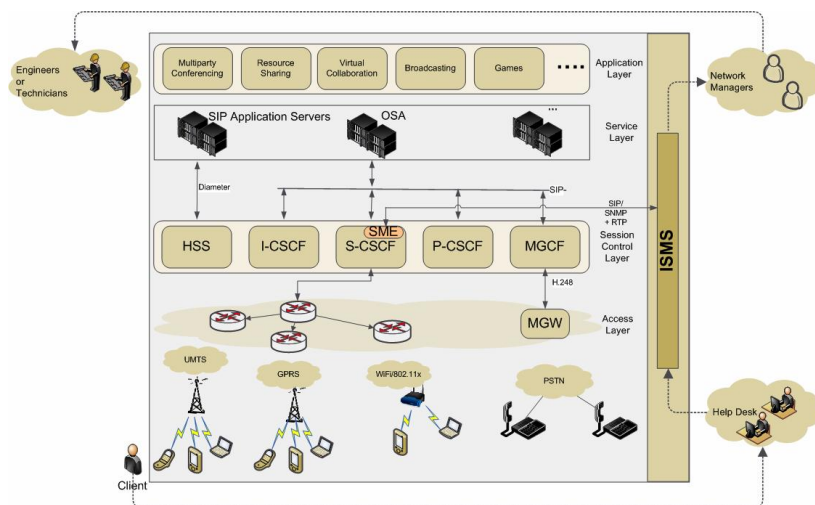


Fig. 2. A simplified IMS environment showing the flow of calls after a service disruption scenario is observed

Service management in the system is handled by the ISMS server. Figure 1-b shows the modular diagram of ISMS. Along with the conventional modules of any management server, there are five more modules for service management. These

modules are SMO Creator, SMO Composer, Threshold Comparer, Error & Fault Detector and Reporter & Notifier.

A typical scenario for how the SMO would help in management of a Service is given in figure 2. A typical service disruption observed by an end user is provided for analyzing the service management. The monitoring and management of this problem can be envisioned as follows:

Service is disrupted while the client is watching IPTV- Customer calls the Help Desk using the VoIP service - From the Help desk the message of Service disruption is sent to ISMS - IPTV_Session_Service_Disruption SMO is created at the ISMS by the SMO Creator module - ISMS sends on-demand request of Managed Objects to the Managed Nodes - Managed Objects (MOs) are sent back by the different IMS, SDP nodes and other network elements - SMO composer builds up SMO from the gathered MOs - Different SMOs provide different measurement of the services, network and quality - Using the SMOs the system can identify the problems in the system by comparing the measured values with the threshold values - Fault detection & localization is performed by the fault detection module - Alarm notification is sent to the configuration management module for reconfigurations - Maintenance parameters are sent to the Network Managers and maintenance personnel - Once the problem is identified, it can also be reported to the concerned nodes (Network Elements) - The notification is made to the service providers and QoS and SLA ensuring nodes - Business processing modules may be notified about the problems in the system - The configuration parameters of those nodes are changed accordingly - Testing can be performed to verify the system after the problem is solved.

5 Conclusion

In this paper, we provided a management server for IMS/SDP solution, which utilizes the new concept of service managed objects. To be a competitor in the field of service provision an IMS/SDP solution needs to be properly managed to provide competitive services at high quality with efficient performance.

References

1. 3GPP, Technical Specification Group Services and System Aspects; IP Multimedia Subsystem (IMS), Stage 2 (Release 7.5.0) (September 2006)
2. Khlifi, H., Gregoire, J.-C.: IMS for Enterprises. *IEEE Communications Magazine* 45(7), 68–75 (2007)
3. Magedanz, T., Witaszek, D., Knuettel, K.: The IMS playground @ FOKUS-an open testbed for generation network multimedia services. In: *First International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities*, February 23-25, 2005, pp. 2–11 (2005)

Evaluating Open Service Access with an Abstract Model of NGN Functions

Samson Lee¹, John Leaney¹, Tim O'Neill¹, and Mark Hunter²

¹ University of Technology, Sydney

² Alcatel-Lucent Australia

Abstract. As new business models and market opportunities are rapidly emerging from the ‘opening up’ of telecommunications networks, we required a better understanding of the effectiveness of using open standards to provide access to functions in NGNs. In this paper we reason about the coverage of openly accessible functions using an abstract model of NGN functionality. Defining and using an abstract model allows us to evaluate the effectiveness of open standards from a perspective where a wide range of NGN functionality can be generalised and conveniently categorised. Subsequently, it will be possible to identify the gaps, which are a subset of functionality that we are specifically interested in for our project.

1 Introduction

In the last several years, new business models and market opportunities have emerged as a result of the ‘opening up’ of next generation telecommunications networks [1]. It is increasingly becoming common for applications to be developed and provided by enterprises outside the traditional network operator domain. The promise is that these applications can utilize the feature-rich service capabilities of the network through standardized Application Programming Interfaces (APIs) such as the OSA/Parlay (Open Service Access) APIs.

As part of research into managing Alcatel-Lucent Australia’s next generation Enriched Experience Network (EEN), we required an abstract model of Next Generation Network functions with which to reason about the coverage of openly accessible functions. The challenge is that the scope of functionality provided by NGNs is massive, and to our knowledge, the coverage of openly accessible functions has generally not been considered in previous publications.

Defining an abstract model of NGN functions allows us to evaluate the effectiveness of open standards from a perspective where a wide range of NGN functionality can be generalised and conveniently categorised. Subsequently, it will be possible to identify the gaps in coverage, which are a subset of functionality that we are specifically interested in for our project. We will use the name, *Separation Abstract Model for Standards-Based Open Networks (SAMSON)* to refer to this model.

The model uses well-known principles of NGNs: the separation of *service* and *transport* layers; and the separation of *management*, *control* and *data* planes.

Combining these two principles of separation, the SAMSON model consists of six functional groupings, which are described next, and used to evaluate the effectiveness of OSA/Parlay open APIs. Specifically for our project, we are interested in providing QoS control functions to applications (application-driven QoS control), and where necessary, we will be focusing on this subset of functionality.

2 Evaluating OSA/Parlay Using the SAMSON Model

We will now apply our SAMSON model as a generalised framework within which to evaluate the effectiveness of OSA/Parlay open APIs in terms of the following six functional groupings.

Service Management-Plane. The OSA/Parlay model provides service life-cycle management functions through a set of APIs called, the OSA/Parlay Framework [2]. Before an application can use the network service capabilities exposed through the OSA/Parlay APIs, it must authenticate itself with the Framework. After authentication, the discovery function enables the application to find out which service capabilities are provided by the OSA/Parlay gateway's APIs. The network services can then be accessed by the methods defined in the OSA/Parlay interfaces.

The OSA/Parlay model provides the following FCAPS functions for managing the services:

Security Management functions are provided as a part of the OSA/Parlay Framework's *Trust and Security Management API*, which includes the authentication and authorisation of services, as well as applications that wish to make use of the services that can be discovered through the Framework. Following successful authentication, the application or service can request access to the other Framework interfaces.

Configuration Management functions are provided as part of the OSA/Parlay Framework's *Registration, Service Discovery and Service Life Cycle Manager APIs*. Before a service can be offered to an application, it must be registered with the Framework. The service is able to register itself to the Framework, which maintains a repository of registered services, only if it has already been authenticated and authorised. Likewise, applications are able to use the discovery function to obtain information on available service capability features only after being authenticated and authorised.

Fault Management is provided as part of the OSA/Parlay Framework's *Integrity Management API*, which is used to inform the application of events that affect the integrity of the Framework, service or client application.

Performance Management functions are provided as part of the OSA/Parlay Framework's *Integrity Management API*, which includes load balancing and heartbeat functions.

Accounting Management functions are provided as part of the OSA/Parlay Framework's *Service Agreement Management and Service Subscription APIs*, as

well as through some Service Capability Feature APIs. The *Account Management API* allows applications to query account balances, retrieve transaction histories, and enable or disable charging-related event notification. The *Charging API* allows applications to charge parties for the services offered.

Service Control-Plane. The OSA/Parlay model plays an important part in the control of the service life-cycle. One of the major goals of the OSA/Parlay model is to enable the creation, deployment, operation and withdrawal of value-added services. OSA/Parlay achieves this by providing API functions for a number of Service Capability Features (SCFs). These SCFs are the packages of network functionality that can be exposed and provided to internal or external applications developers, who can utilize the wide range of capabilities to create innovative new applications for customers.

Some of the SCFs that have already been standardised by ETSI and 3GPP include: Call Control; User Interaction; Mobility (including user location and user status); Terminal Capability; Generic Messaging; Policy Management; Presence and Availability Management; Directory/User Profile; Connectivity Management; and Data Session Control. For more information on these SCFs, refer to the specifications.

We are interested in the Connectivity Management SCF, which provides QoS control to applications. Application-driven QoS control is a function of the service control-plane because it is controlled directly by an application, and provides added-value for the customer. Application-driven QoS control may require transport control-plane functions such as functionality provided by Diffserv, RSVP, etc.

Service Data-Plane. OSA/Parlay does not explicitly provide service data-plane functions, since it is not within the scope of the OSA/Parlay goals. However, implementing high-level API functions implicitly require lower-level data-plane functions such as media encoding/decoding, encryption/decryption, etc.

Transport Management-Plane. Traditionally, the management of networks has been limited mainly to connectivity. In the past, Operational Support Systems (OSS) and Network Management Systems have focused on monitoring, alarms and basic control of network devices for transport related functions.

OSA/Parlay does not explicitly provide service management-plane functions, since it is not within the scope of the OSA/Parlay goals. However, in practice, devices such as Alcatel-Lucent's 8601 OSA/Parlay Gateway provide interfaces for OSS integration in addition to the open APIs for application development.

Transport Control-Plane. Due to the separation of service and transport, the OSA/Parlay models rely on the assumption that there is basic connectivity (e.g. an IP network). This basic connectivity (routing achieved through OSPF, etc), is independent of the OSA/Parlay models.

For more advanced transport functions, the OSA/Parlay models provide a high-level API for applications to utilize the underlying service capabilities of

the transport network. In order for OSA/Parlay to implement these high-level API functions, they need to be mapped to underlying transport control-plane.

For example, the Connectivity Manager provides a simple API for applications to control QoS, which could be realised by executing SNMP messages in the transport control-plane. The specific method in the transport control plane is generally not prescribed.

Connectivity Manager satisfies this requirement by presenting an abstraction of the network capabilities to applications. It is possible to translate these abstracted functions into specific mechanisms implemented by the access network.

The specific mechanisms implemented by the access network resides in the transport control and data planes of the SAMSON abstract model since the signalling protocols by themselves have little direct value to the customer, and the per-packet processing including queueing has no intrinsic meaning.

Transport Data-Plane. OSA/Parlay does not explicitly provide transport data-plane functions, since it is not within the scope of the OSA/Parlay goals. However, implementing high-level API functions implicitly require lower-level data-plane functions such as per-packet forwarding/processing, etc.

3 Summary and Conclusions

We gave an overview of using the SAMSON abstract model to evaluate the coverage of OSA/Parlay, demonstrating that OSA/Parlay offers good coverage of NGN functionality related to the service management plane. It does this by providing the OSA/Parlay Framework interfaces, which expose discovery, authentication and other management functions to internal as well as third party applications. However, the integration of OSA/Parlay with existing OSS and NMS is implementation (product) dependent. OSA/Parlay does not provide sufficient coverage of functions related to application controlled Quality of Service in the service control-plane.

OSA/Parlay is an important part in enabling innovative new services because it moves the complexity of the underlying network into a high-level API. We will continue our evaluation of Open Service Access using our abstract model of NGN functions as a part of future work.

References

1. Moerdijk, A.J., Klostermann, L.: Opening the Networks with Parlay/OSA: Standards and Aspects Behind the APIs. IEEE Network, 58–64 (2003)
2. 3GPP: Open Service Access (OSA); Application Programming Interface (API) Framework (Release 5) v5.1.0. Technical Specification Group Core Network (2002)

Methods for Rapidly Testing Node Reachability with Congestion Control and Evaluation

Naoki Tateishi, Mitsuho Tahara, Yu Miyoshi, and Souhei Majima

NTT Network Service Systems Laboratories, NTT Corporation
9-11, Midori-Cho 3-Chome, Musashino-shi, Tokyo, 180-8585 Japan
{tateishi.naoki, tahara.mitsuho, miyoshi.yu,
majima.souhei}@lab.ntt.co.jp

Abstract. Most ISPs and telecom carriers test reachability to their nodes and interfaces by SNMP or ICMP echo. However, testing for on the order of 10,000-100,000 nodes by traditional techniques takes a long time. Furthermore, huge numbers of short packets occupy the capacity of the links and interfere with other flows. At this time, test software using traditional techniques cannot distinguish whether the reason for packet loss is unreachability or congestion. We should use methods for sending many packets, a congestion detection method, and congestion control method when we test for node reachability. By using these methods, ISP and telecom carriers will obtain reliable results. In this paper, we report such methods and experimental results of this implementation.

1 Introduction

Generally, ISPs and telecom carriers test their networks using SNMP [1] and ICMP [2]. However, testing node reachability in huge networks may take many hours. So decreasing the time between a node failure and its detection is difficult [3]. Thus, we consider techniques for testing node reachability over huge scale networks.

Operators in ISPs and telecom carriers ping when they test node reachability in their networks because congestion over links causes packet loss and affects results. Sending multiple pings prevent mistakes. However, this method has the following problems

- The decision lacks valid reason. We cannot detect congestion using this method.
- Sending multiple pings can cause congestion.

We do not know how many pings are required to obtain a reliable result.

Sending multiple pings would escalate congestion and increase the packet loss rate. Furthermore, this method would affect other flows, such as data, voice, and video..

To address these problems, we also consider load distribution, congestion detection, and congestion control. We should distinguish packet losses from failed nodes in node reachability tests. In addition, if we recognize that the reason for packet losses is congestion, we can decrease the ping rate. Hence, we define requirements for the ping test method below.

- Agent can test reachability of over 500,000 nodes within 1 minute.
- Agent can prevent congestion over links during transmission of short packets.

Therefore, two methods, one for sending huge numbers of pings and one for controlling congestion, must be considered. In particular, we must prevent these methods from affecting other flows during node reachability tests.

We propose methods for sending a huge numbers of pings, distributing traffic, and preventing congestion. Then, we evaluate the effectiveness of these methods.

2 Method to Send Huge Numbers of Pings

In this section, we explain the method for sending a huge numbers of pings. First, we introduce the conventional implementation of a ping program and our improved program. Second, we show the results of evaluations.

2.1 Proposed Method

Programs for sending pings are implemented in most operating systems. In most programs, one process can send pings to one destination. The conventional implementation is shown in Fig. 1(a). However, these programs are problematic to use in huge scale networks. We can only execute on the order of 100 processes simultaneously because many operating systems only permit the execution of on the order of 100 processes. In addition, we cannot shorten intervals for sending pings because ping programs wait for the arrival of pings until a timeout occurs.

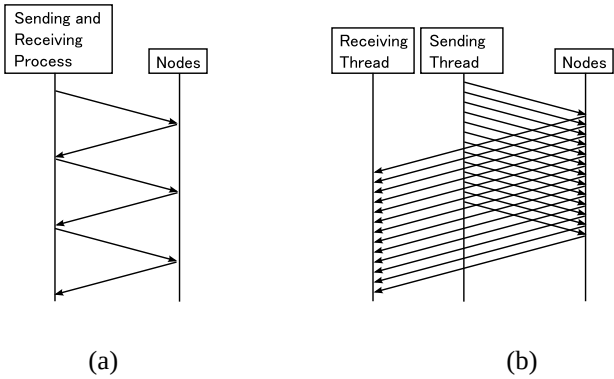


Fig. 1. (a) Conventional implementation (b) Proposed implementation

We solved these problems by implementing software with new functions. The proposed implementation is shown in Fig. 1(b). We implemented one thread to send packets and another thread to receive packets. We sent on the order of 10,000 packets at a faster pace by parallelizing the sending and receiving threads so that they are simultaneous.

2.2 Evaluation

To test the effectiveness of this implementation without the effects of delay and link congestion, we built an ideal environment. We connected two computers to a gigabit router. We set no bandwidth limitations.

We sent 500,000 pings by the proposed method in 17 seconds, so we can ping on the order of 10,000 in a second. We can only ping on the order of 1,000 in a second using a traditional implementation [3].

3 Load Distribution and Congestion Control

In the previous section, we showed methods for pinging a huge number of times. When we use this method, we should consider load distribution and link congestion control to avoid affecting other user traffic. In this section, we introduce a load distribution method and congestion control method. Then, we show results of the evaluations of the congestion control method.

3.1 Load Distribution

In a large-scale network, we often divide IP addresses into subnets, which are then allocated to small parts of the network to simplify management. If pings are sent in order of IP address, many pings are transmitted on the links and nodes to one subnet. Hence, this method could cause heavy congestion.

We propose a load-balancing method to avoid congestion. We decide which nodes ping in a random manner. If we use the proposed methods, the order in which the pings are performed is decided randomly. The load on the links in a large-scale network will be averaged by using this method.

3.2 Detection of Congestion and Control of Sending Rate

We showed a load-balancing method that can decrease the probability of congestion occurring. However, this method cannot completely avoid congestion, so we propose another method for detecting congestion and controlling the transmission rate to ensure that network management traffic does not affect customer traffic.

3.2.1 Proposed Method

We surveyed current methods of ping congestion control but we could not find suitable ones because, in the past, pings did not need to be sent over huge scale networks, and network congestion was not a concern. We also surveyed TCP congestion control schemes. However, we found a problem with the schemes. If the data transmission does not work properly under the environment, the receiving node does not work properly.

We developed the congestion control method described below to solve these problems. In this method, pings are performed at the highest set rate. Then, we ping again to nodes from which we did not receive replies. If we get replies after pings are re-transmitted, we identify this as an indication of congestion. This is because if there

was no congestion, we would have received replies from all nodes in operation after the original pings. If we detect congestion, we decrease the transmission rate.

3.2.2 Evaluation

The result of the evaluation under a bandwidth constraint is shown in Fig. 2. We connected computers A and B to a gigabit router. We set the bandwidth constraint to 5 Mbps, then we pinged 500,000 times. If the flow bandwidth exceeded the constraint, packets were dropped. The bandwidth exceeded 5 Mbps after the test started. The agent detected congestion and adjusted the bandwidth. If we used the TCP rate control scheme, the packet transmission rate will drop.

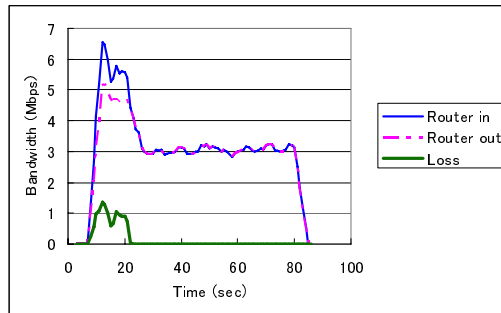


Fig. 2. Ping sending rate with upper bandwidth constraint

4 Conclusion

We conclude from our evaluations that the proposed implementation of a ping program enables us to send and receive a huge number of packets without packet loss. In addition, we effectively tested the node reachability of large scale networks.

In the ping congestion control method, the bandwidth would decrease if we use traditional TCP methods under these environments, for example, most nodes do not work properly and packet losses occur frequently. However, the proposed method solved these problems and maintained the transmission rate.

We will test the effectiveness of these methods in other networks, and consider test schemes for multi-protocols used in large-scale networks.

References

1. Case, J., Fedor, M., Schoffstall, M., Davin, J.: The Simple Network Management Protocol, STD 15, RFC 1157 (May 1990)
2. Postel, J.: Internet Control Message Protocol, STD 5, RFC 792 (September 1981)
3. Fping, <http://fping.sourceforge.net/>

The Design of an Open and Integrated Sensor Network Management Platform

Michalis Kalochristianakis¹, Vasileios Gkamas¹, Georgios Mylonas¹,
Sotiris Nikolettseas¹, Jose Rolim², and Emmanouel Varvarigos¹

¹Department of Computer Engineering and Informatics, University of Patras, Greece

²Department of Computer Science, University of Geneva, Switzerland

{kalohr, vgkamas, mylonasg, nikole, manos}@cti.gr, rolim@cui.unige.ch

Abstract. We present the conceptual basis and the initial planning for an open source management architecture for wireless sensor networks (WSN). Although there is an abundance of open source tools serving the administrative needs of WSN deployments, there is a lack of tools or platforms for high level integrated WSN management. The current work is, to our knowledge, the first effort to conceptualize and design a remote, integrated management platform for the support of WSN research laboratories. The platform is based on the integration and extension of two innovative platforms: jWebDust, a WSN operation and management platform, and OpenRSM, an open source integrated remote systems and network management platform. The proposed system architecture can support several levels of integration in order to cover to multiple, qualitatively differentiated use-cases.

Keywords: Remote management, wireless sensor networks, pervasive communications, open source.

1 Introduction

The effective and flexible integration of sensory systems with interconnection networks and grid technologies is a critical step towards developing pervasive computing and communication services and applications. Because of sensor network inherent characteristics such as heterogeneity, limited bandwidth and energy constraints, WSN management and monitoring architectures need to: (i) provide a wide range of services so as to cover the whole spectrum of supported WSN architectures, (ii) minimize the overall WSN management effort and make it uniform to the degree possible, (iii) reduce the needs for network administration and sensor node software management and (iv) expose web interfaces for WSNs management. A number of administrative and management tasks that are typically supported by standard management tools for workstations would be also valuable to WSNs operators and administrators. Such tasks are: multiple WSNs management and monitoring, web-based control, remote command execution or remote configuration, software / firmware upgrade, reporting and grouping.

This paper presents an architecture that is capable of delivering remote management functionality to wireless sensor networks. The idea is based on the combination of

jWebDust [5], a software environment that allows the implementation of customized applications for wireless sensor networks, and OpenRSM [8]. In order to bring management to WSN, it is essential to automate routine WSN practice. jWebDust provides all standard WSN management functionality, including querying, monitoring, data logging and visualization. jWebDust employs an extendable architecture and provides easy interfacing and API. OpenRSM provides the implemented distributed logic suitable to deliver services, such as inventory and asset management, software delivery, remote control and network monitoring, all integrated in one environment.

The paper is organized as follows. Section 2 describes the related state of the art on monitoring and administration tools for wireless sensor networks. Section 3 presents the proposed integrated architecture and section 4 presents the integration methodology. Conclusions and future work are discussed in Section 5.

2 Related Work

Software environments that provide the necessary tools and operations to allow the monitoring and administration of a wide range of WSN's applications are relatively few, and to our best knowledge, none of them integrates all the desired management functionality in a single environment.

TinyDB [2] is an example of an application that allows multiple concurrent queries, event-based queries and time synchronization through an extensible framework that supports adding new sensor types and event types. MoteWorks [6] is a commercial WSN management product. It is based on an n-tier architecture model and offers standard management functionality and it supports a significant number of sensor node hardware platforms. MoteWorks also offers visualization tools and login services [1]. ArchRock [7] is a commercial product that offers WSN management capabilities. It offers sensing functionalities and APIs to interface with, and integration with IP networks, using a 6LoWPAN network stack inside the sensor network. TWIST [3] and MoteLab [4] are examples of testbed deployment management environments, targeted toward research teams. They provide capabilities such as job scheduling and network sniffing designed for special-purpose environments that focus on WSN applications testing.

The above platforms and tools provide administrative functionality, mostly in terms of development. Their goal is not to deliver an integrated remote management WSN environment capable to convey detail and mass administration and monitoring commands to the wireless sensor network.

3 The Concept of Integrated WSN Management

jWebDust and OpenRSM can complement each other in order to deliver integrated management to WSNs. With respect to the level of integration we distinguish the following management layers: (i) WSN server / gateway infrastructure management, (ii) WSN server functionality support, application programming interface (API) integration and (ii) firmware management.

OpenRSM can be primarily used to deliver infrastructure management to WSN servers that is, bring inventory and assets management, network monitoring, software delivery, and remote desktop control to laboratory servers. This level of integration is achieved by incorporating the OpenRSM agent within a jWebDust installation.

OpenRSM can be further integrated with jWebDust. In particular, current work focuses on extending OpenRSM to support jWebDust services, APIs and tools and to expose them as special-purpose remote management services. The result of the integration will be that OpenRSM will be capable to schedule and synchronize the execution of jWebDust-specific jobs. Users will then be capable to define custom jWebDust-specific jobs corresponding to distinct administrative task manageable within the jWebDust platform. OpenRSM will embody standard jWebDust template jobs. Users will be capable to utilize the template jobs to create custom ones. Jobs can also be correlated with machines and then be submitted to the OpenRSM integration server; the server will forward them to the agent running at WSN's server/gateway where jobs will be executed as jWebDust procedures.

In order to cover advanced needs for WSN management, such as firmware development and deployment, a higher level of integration is required. Firmware construction, deployment and discovery of individual sensor nodes require firmware level integration that will enable the management platform to deliver identifiable sensor nodes, capable to dynamically change their state and runtime environment. It is therefore necessary for the management platform to support the underlying development environment and to include a firmware construction framework. The underlying development environment for sensor node firmware, TinyOS [9], offers development tools that can be integrated into custom OpenRSM jobs. TinyOS functionality can be used by OpenRSM via TinyOS job templates, disposable to users for customization or immediate usage. The compilation and deployment tasks can be formalized to produce accurate and safe results, disallowing the creation of modules that may result to WSN malfunction. For example, incompatibilities in the discovery or firmware update services must be eliminated via the formalization of TinyOS jobs. The output of the TinyOS jobs will be registered and stored.

4 Integration Methodology

The integration of the two platforms is currently in progress. An early milestone is setting jWebDust under OpenRSM management by producing a custom agent module for WSN gateways. The agent module can be integrated into the codebase of the jWebDust platform or it can be deployed as an external module. The next milestone is integrating the base of high level system management functionality for the specific needs of WSN gateways, into the agent module. Integration will produce agent connectors for the jWebDust API interface. After this point the integration will proceed to all OpenRSM levels and eventually to the management console where the user will be presented with visual representations of the managed boards and even sensors. At this point, the integration of jWebDust and OpenRSM may even consider branching from OpenRSM and jWebDust, since it will have the potential needed for building end-user services for WSN management.

5 Conclusions and Future Work

This paper presents the conceptual design of an integrated management environment enhanced with WSN management functionality. The design is based on the integration of the WSN monitoring platform jWebDust with the OpenRSM systems and network management system. The derived platform is an integrated remote management WSN environment, under which the overall administration and monitoring of the wireless sensor network can be performed both in low and high level. Current work includes the gradual integration of the two platforms in accordance with the integration levels, as mentioned in section 4, and the evaluation of its feasibility and performance (implementation overhead, evaluation of performance metrics).

Acknowledgements

This work has been partially supported by the IST Programme of the European Union under contract number IST-2005-15964 (AEOLUS), the Programme under the European Social Fund (ESF) and Operational Program for Educational and Vocational Training II (EPEAEK II) and the Programme PENED of GSRT under contract number 03ED568.

References

1. Mote-VIEW monitoring software, Crossbow Technology Inc., <http://www.xbow.com/>
2. TinyDB: A declarative database for sensor networks, <http://telegraph.cs.berkeley.edu/tinydb/>
3. TWIST Community Web Site, <http://www.twist.tu-berlin.de/wiki>
4. MoteLab, Harvard Experimental Wireless Sensor Network Testbed, <http://motelab.eecs.harvard.edu/>
5. Chatzigiannakis, Mylonas, G., Nikoletseas, S.: jWebDust: A java-based generic application environment for wireless sensor networks. In: The proceedings of the 1st International Conference on Distributed Computing in Sensor Systems (DCOSS), 2005, pp. 376–386 (2005); Also, in the International Journal of Distributed Sensor Networks (IJDSN) (to appear, 2008)
6. Mote-Works monitoring software, Crossbow Technology Inc., <http://www.xbow.com/>
7. ArchRock, <http://www.archrock.com/>
8. Karalis, Y., Kalochristianakis, M., Kokkinos, P., Varvarigos, E.: OpenRSM: An Open Source Lightweight Integrated Remote Network and Systems Management Solution, under publication in International Journal of Network Management
9. TinyOS, <http://www.tinyos.net/>

Self-organized Cluster Based Multi-hop Routing for Wireless Sensor Networks

Hongjoong Sin, Sungju Lee, Jangsu Lee, Seunghwan Yoo, Sanghyuc Lee,
Jaesik Lee, and Sungchun Kim

Department of Computer Science and Engineering, Sogang Univ.

CPO Box 1142, Seoul, Korea

siceman00@gmail.com, {golgon, jangso417}@hotmail.com,

ossforever@sogang.ac.kr, ddt101@lycos.co.kr,

ljsljslove@hotmail.com, ksc@sogang.ac.kr

Abstract. It is difficult to apply other criteria techniques into wireless sensor networks. Since, sensor nodes are generally very small and have scarce resources. The energy efficient routing to prolong the life time is the major research issue in wireless sensor networks. Many researches proposed cluster-based schemes to reduce energy dissipation. These schemes have to maintain the information of overall nodes or create an imbalanced cluster formation. To solve these problems, we propose a self-organized clustering based on the multi-hop routing. In our proposed scheme, clusters are organized efficiently through the extension of clusters in distributed way without any location data. According to performance evaluation using NS-2, the proposed scheme increase the network lifetime longer than the previous schemes.

Keywords: Wireless Sensor Networks, Multi-hop Routing, Clustering.

1 Introduction

The wireless sensor networks consist of a number of small sensor nodes scattered randomly in the sensing area. Due to the limited resources of the sensor nodes, it is extremely important to develop energy efficient schemes. Recently many researchers focus on the energy efficient algorithms to improve performance of WSN. In [1], Low-Energy Adaptive Clustering Hierarchy(LEACH) is proposed. LEACH reduces a number of transmissions for each node. But Cluster Header(CH) consumes much energy for communication in LEACH. Many researchers propose various techniques to prolong the network lifetime using cluster based schemes [2]~[7].

We propose a self-organized cluster based multi-hop routing scheme in this paper. To solve imbalance problem in clustering, clusters in our scheme are extended automatically in distributed way. After clustering, sensor nodes transfer sensing data to the base station(BS) through neighbor nodes using cluster information.

2 Proposed Protocol

In this section, we will describe our self-organized cluster based multi-hop routing scheme. We assume that all sensor nodes are synchronized with BS and sensor nodes

are homogeneous in our scheme. There are 4 types of node states, Initial, CH, Member, Border in our scheme. Our scheme consists of self-organized clustering and multi-hop routing. Detailed descriptions of our proposed scheme are as follows.

2.1 Clustering

In Initialize Phase, the Base Station(BS) synchronizes with all sensor nodes. The BS broadcasts an initialization message which includes random waiting times per nodes to all nodes. When a node receives an initialization message, it relays that message to its neighbor nodes and set up its state as Initial. Each node calculates a distance from the BS to itself using a hop count of the initialization packet.

In Cluster Formation Phase, each node decreases its waiting time synchronized with the timer of the node. When the waiting time becomes 0, the node determines itself to be the CH. The CH broadcasts a Hello message to invite neighbor nodes. When the node state of neighbor nodes which receive that message is Initial, they change their node state as Member and rebroadcasting a Hello message to extend a cluster size. When neighbor node is included with different clusters, it changes its state as Border and maintains routing table to connect each cluster, stop extending the cluster size.

Clusters automatically extend their coverage by some rules. The BS doesn't have to know the information of all nodes. Clusters organize its formation in distributed manner. This mechanism prevents CH constructing imbalanced cluster formations. Each node sends a Hello message only one time. This scheme reduces clustering overheads.

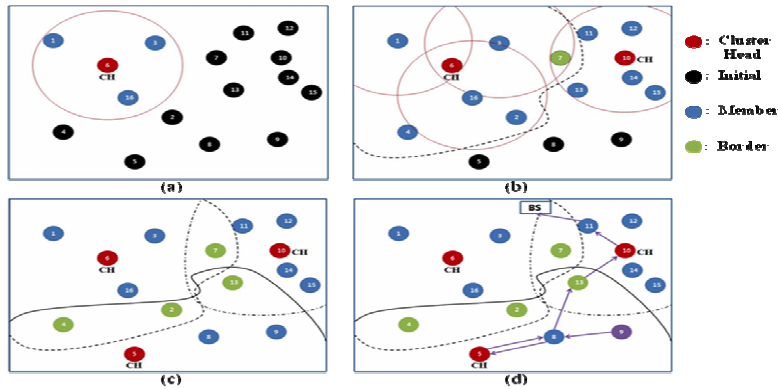


Fig. 1. (a)~(c) Example of Cluster formation, (d) Multi-hop Routing

2.2 Multi-hop Routing

When a node has a packet to deliver to the BS, it sends a packet to its CH. After sending, CH selects a Border node to relay a packet. The formula (1) shows how to select a Border node.

$$[(1 - \alpha) \times (1 - \frac{H_{MEM}}{H_{MAX}})] + [\alpha \times (\frac{E_{MEM}}{E_{MAX}})] \quad (1)$$

The H_{MAX} is maximum hop count from the BS of the WSN, the H_{MEM} is hop count value of the node. The E_{MAX} is an initial energy levels, the E_{MEM} is remaining energy of the node. The α is a value between 0 and 1. It reflects the extent to which distance and energy characteristics are considered for intermediate node selection. Using (1), the CH select an appropriate node between Border nodes by consider energy and distance. Then the CH transmits packets to Border node, the Border node relay packets to Neighbor CH. Multi-hop routing scheme can reduce energy consumption for transmission. Because energy consumption for transmission increases in proportion with (distance)².

3 Performance Evaluation

We have implemented our proposed scheme in NS-2(version 2.31). For measuring, we have compared our proposed scheme with existing schemes LEACH[1] and LNCA[7].

In Fig. 2. (a), we compare the average remaining energy of our proposed scheme to existing schemes with 50 nodes. The remaining energies of LEACH and LNCA are rapidly decreased. But our proposed scheme achieves better performance than LEACH and LNCA. In our scheme, a clustering overhead is smaller than LNCA and the CH selects the best node to relay packets. The multi-hop routing reduces a waste of energy. Fig. 2. (b) shows an average network lifetime comparison between LEACH, LNCA and our proposed scheme with different number of nodes. The performance of LNCA is lower than our proposed scheme when the number of nodes is larger.

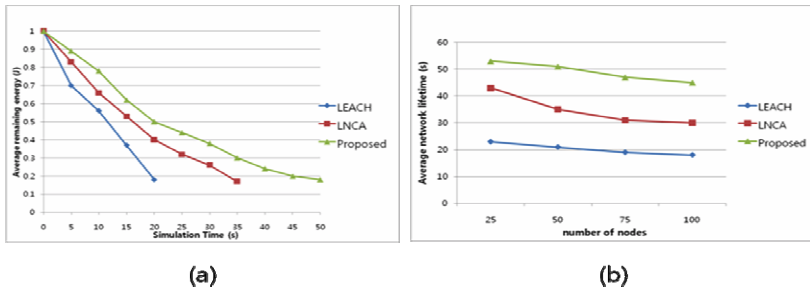


Fig. 2. (a) Average Remaining Energy (50 Nodes), (b) Average Network Lifetime Comparison

4 Conclusion

In the paper, we propose a self-organized clustering which is used for multi-hop routing scheme in wireless sensor networks. Our scheme selects a CH randomly and extends cluster size in distributed way. In this way, we prevent generating imbalanced

cluster formation and provide scalability. And then proposed routing scheme find a best path using cluster information.

The experiments show that our scheme improves the overall energy efficiency compared with previous schemes. And the whole throughput of the network is also increased.

Our future work will focus on improving the cluster formation scheme more detail. The details are like this; decision of maximum cluster size, CH reselection, mediation between clusters by merging or split.

Acknowledgments

This work was supported by the Brain Korea 21 Project in 2008 and the Korea Research Foundation Grant funded by the Korean Government (MOEHRD, Basic Research Promotion Fund) (KRF-2006-331-D00475).

References

1. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy-Efficient Communication Protocol for Wireless Microsensor Networks. In: Proceedings of the Hawaii International Conference on Systems Sciences, vol. 2, pp. 10–19 (2000)
2. Heinzelman, W.: Application-Specific Protocol Architectures for Wireless Networks. PhD thesis, Massachusetts institute of technology (2000)
3. Zhang, W., Liang, Z., Hou, Z., Tan, M.: A Power Efficient Routing Protocol for Wireless Sensor Network. In: Proceedings of the London International Conference on Networking, Sensing and Control, pp. 15–17 (2007)
4. Luo, H., Ye, F., Cheng, J., Lu, S., Zhang, L.: TTDD: Two-tier Data Dissemination in Large-scale Wireless Sensor Networks. In: ACM/Kluwer Mobile Networks and Applications (MONET), Special Issue on ACM MOBICOM (2003)
5. Xu, Y., Heidemann, J., Estrin, D.: Geographhy-informed Energy Conservation for Ad-Hoc Routing. In: Proceedings of the Seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking (2001)
6. Lin, C., Chou, P., Chou, C.: HCDD: hierarchical cluster-based data dissemination in wireless sensor networks with mobile sink. In: International Wireless Communications and Mobile Computing Conference (IWCMC), Vancouver, Canada, pp. 1189–1194 (2006)
7. Xia, D., Vlajic, N.: Near-optimal Node Clustering in Wireless Sensor Networks for Environment Monitoring. In: International Conference on Advanced Information Networking and Applications (AINA 2007), Niagara Falls, Canada (2007)

An Adaptable Method of E-Workflow Composition Based on Distributed Awareness

Hongbin Sun^{1,2} and Yongsheng Ding²

¹ School of Electrical Engineering and Information , Changchun Institute of Technology,

² College of Information Sciences and Technology, Donghua University, 201620, Shanghai
hbsun@mail.dhu.edu.cn

Abstract. A method of E-workflow composition based on distributed awareness is proposed. The mobile agent represents E-service and constitutes an autonomic unit. The E-workflow composition is achieved by the evolution of the mobile agent based on the semantic negotiation. By understanding properties of messages, the mobile agent use projection-join closure to capture such message situations, the optimization of the E-workflow composition is translated into dynamic services selection with QoS global optimization. We propose an improved algorithm based on quick sort and NSGA-II to resolve it. The simulation results show that the approach can significantly improve the E-workflow performance. It adapts well to the changes of dynamic environments.

Keywords: E-workflow composition, adaptable, mobile agent, distributed awareness.

1 Introduction

With the explosive growth of Internet, more enterprises are providing various E-services. Workflow management systems are capable of integrating business objects for setting up E-services in an amazingly short time and with impressively little cost [1]. Based on the mobile agent architecture and service composition platform [2], we present the novel method of E-workflow composition based on the design of distributed awareness.

2 The Method of E-Workflow Composition Based on Distributed Awareness

As an atomic unit of E-workflow composition, the aware agent (mobile agent) includes three modules. Attributes describe the characteristic of an agent itself. Function is designed to evaluate the matching ability of the message to the other mobile agents. Behavior contains interface operation, information issue, and energy transmission. The aware agent is a unit developed by java. Different aware agents may contribute to different E-services. We present the method of service discovery based on the message matching [3, 4].

A matching message is exchanged among aware agents to achieve service matching for composition E-workflow. $ES1(Pr1, SI_{ES1}, QoS1)$ represents the request of matching service with an output message, and $ES2(Pr2, SI_{ES2}, QoS2)$ represents E-service with input message of aware agent. The matching strength of message $SS_{bio-entity}^{ws}(ES1, ES2)$ is defined as the matching ability.

$$SS_{bio-entity}^{ws}(ES1, ES2) = (1 - \eta) \cdot MS_{bio-entity}^{ws}(SI_{ES1}, SI_{ES2}) + \eta \cdot SI_{bio-entity}^{ws}(Pr1, Pr2) \quad (1)$$

Where $SI_{bio-entity}^{ws}(Pr1, Pr2)$ is two property's similarity, $MS_{bio-entity}^{ws}(SI_{ES1}, SI_{ES2})$ is the interface similarity. η is the value of weight. A threshold of the matching strength of message λ will be specified $\lambda (0 < \lambda < 1)$. If $SS_{bio-entity}^{ws}(ES1, ES2) > \lambda$, it succeeds for matching relation directly between the aware agents.

$MS_{bio-entity}^{ws}(SI_{ES1}, SI_{ES2})$ is defined as the matching ability to E-service interface message. $SM(C_1, C_2)$ is given by the similarity of between SI_{ES1} and SI_{ES2} , which relies on the concept similarity C_i and C_j in ontology. β is a constant and can be set to the distance when the similarity of two words is equal to 0.5.

$$MS_{bio-entity}^{ws}(SI_{ES1}, SI_{ES2}) = \frac{1}{n} \left[\sum_{j=1}^n \max_{i=1}^m (SM(C_i, C_j')) \right] \quad (2)$$

$$SM(C_1, C_2) = \begin{cases} 1 & C_1 = C_2 \\ SM_{con}(C_1, C_2) & C_1 < C_2, C_1 > C_2 \\ 0 & C_1 \neq C_2 \end{cases}$$

$$SM_{con}(C_1, C_2) = \frac{\beta \times (L_1 + L_2)}{(Dis(C_1, C_2) + \beta) \times \max(|L_1 - L_2|, 1)}$$

$$MS_{bio-entity}^{ws}(SI_{ES1}, SI_{ES2}) \in [0, 1]$$

$$SI_{bio-entity}^{ws}(Pr1, Pr2) = \frac{\max \sum_{i=1}^m \sum_{j=1}^n SM_{con}(C_i^{Pr1}, C_j^{Pr2}) + \max \sum_{i=1}^p \sum_{j=1}^q SIM_{pro}(P_i^{Pr1}, P_j^{Pr2})}{\min(m, n) + \min(p, q)} \quad (3)$$

$$\text{Where } SM_{con}(C_i^{Pr1}, C_j^{Pr2}) = \frac{\beta \times (L_i + L_j)}{(Dis(C_i^{Pr1}, C_j^{Pr2}) + \beta) \times \max(|L_i - L_j|, 1)}$$

$$SIM_{pro}(P_i, P_j) = \begin{cases} 1 & P_i = P_j \\ \frac{1}{Dis(P_i, P_j)} & P_i < P_j, P_i > P_j \\ SM_{con}(P_i, P_j) & P_i \neq P_j \end{cases}$$

E-service quality depends on four factors: Latency (L), Availability (A), Trust (T), and Cost (C). We propose the definition of multi-objective optimization problem for the workflow QoS attributes as formula (4). $Mep(Q)$ is the value to evaluate global QoS for one attribute $\overrightarrow{Q_{attribute}^{all}}$, $L^k(x)$, $A^k(x)$, $T^k(x)$ and $C^k(x)$ are constraints of

E-service in the composition of the workflow, $\mu \in [0,1]$ is a weight to global constant $Mep(Q)$, $MepT_0$, $MepA_0$, $MepH_0$, $L_{\max}^0$, $A_{\max}^0$, $T_{\max}^0$, $C_{\max}^0$ represents the threshold of criterion. The global QoS algorithm of $Mep(Q)$ used the abstract reduction rules for sequential, choice and loops systems to evaluate global QoS.

$Min(Mep(Latency), Mep(Cost))$

Subject

$$\left\{ \begin{array}{l} Mep(Trust) \geq MepT_0 \\ Mep(Availability) \geq MepA_0 \\ \mu \cdot Mep(Trust) + (1-\mu) \cdot Mep(Availability) \geq MepH_0 \\ L^k(x) < L_{\max}^0 \quad k \in K \\ A^k(x) > A_{\max}^0 \quad k \in K \\ T^k(x) > T_{\max}^0 \quad k \in K \\ C^k(x) < C_{\max}^0 \quad k \in K \end{array} \right. \quad (4)$$

In this paper, the novel algorithm (QK-MOWE) based on the improvement quick sort [5] and NSGA-II algorithm is proposed. It can be proved that the time complexity of Algorithm NDSet () is less than $O(n \log n)$. It is better than $O(n^2)$ in the NSGA II. Suppose that the size of evolutionary population P is n , and P_t is t -th generation of the population. Q_t is a new evolutionary population from P_t that is updated by the selection, crossover and mutation operators, and the size of Q , is also n . Let $R_t = P_t \cup Q_t$, and the size of R_t is $2n$. The non-dominated set P_1 is generated from R_t , with the quick sort procedure. If $|P_1| > n$, the clustering procedure is used to reduce the size of P_1 , and to keep the diversity of P_1 at the same time. The size of P_1 will be n after the clustering process.

3 Simulation and Discussion

The simulation experiment is constructed on Windows 2000 operation system with Intel Pentium 4 processor (2.4 GHz) and 512 MB RAM. The results of performance experiments are as shown in Fig.1.

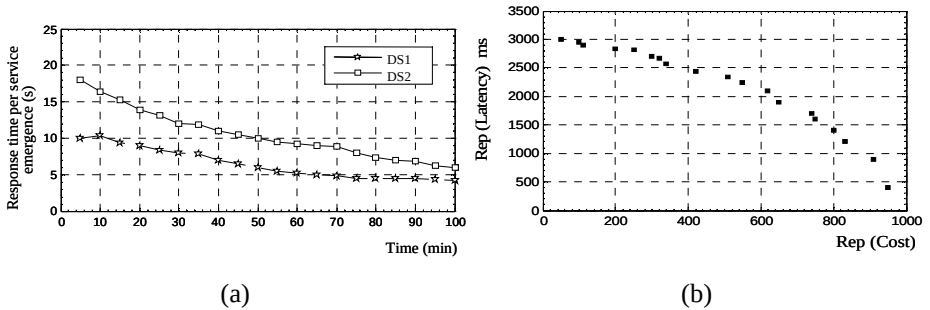


Fig.1. Response time and the Pareto optimal front in DS1

The response time per service composition represents the efficiency. We give out the preference measurement with 100 and 200 common service resources (Fig.1.a). At first, response time is 18 s and 10 s. Individuals will cluster the social world through the long-term collaborations. In our approach, aware agents with related service information will be located nearby one another in the relationship network, allowing quick location of corresponding services, which results into clustering of aware agents with related service resource information. Obviously, aware agent gradually obtains many relationships to them, leading to improved performance in discovery process. The response time is about 5 s and 6.3 s, which achieves the minimum value.

The size of the population is DS1 (100), the crossover and mutation probabilities of $P_c=0.9$ and $P_m=0.1$ are used. The size of the population is 100. Fig.1.b shows the distribution of the Pareto optimal solutions with 19 individuals, and it costs 2.62 s to run 200 generations of QK-MOWE. Note that a Pareto-front represents a gradient of solutions and is better suited for the end-user.

4 Conclusions and Further Work

Aware agents may establish relationship with interaction partners to provide E-workflow. Based on the message matching, the optimization of the E-workflow composition is translated into dynamic services selection with QoS global optimization. The next work is on issue about improving the efficiency of the message matching. It is necessary to reduce the search space during problem solving. In addition, more experiments will be designed to evaluate the method in E-workflow composition and management.

Acknowledgments

This work was supported in part by the Key Project of the National Nature Science Foundation of China (No. 60534020), Program for New Century Excellent Talents in University from Ministry of Education of China (No. NCET-04-415).

References

1. Zhuge, H.: The Future interconnection environment. *IEEE Computer* 38(4), 27–33 (2005)
2. Ding, Y.S., Sun, H.B., Hao, K.R.: A bio-inspired emergent system for intelligent Web service composition and management. *Knowledge-Based Systems* 20(5), 457–465 (2007)
3. Cardoso, J., Lenic, M.: Web Process and Workflow Path mining using the multimethod approach. *International Journal of Business Intelligence and Data Mining* 1(3), 304–328 (2006)
4. Zeng, L., Benatallah, B., Ngu, A.H.H., Dumas, M., Kalagnanam, J., Chang, H.: QoS-aware middleware for Web services composition. *IEEE Trans. Software Engineering* 30(5), 311–327 (2004)
5. Zheng, J.H., Ling, C.X., Shi, Z.Z., Xie, Y.: Some discussions about MOGAs: Individual relations, non-dominated set, and application on automatic negotiation. *Evolutionary Computation* 1, 706–712 (2004)

An Adaptive Control Scheme of Reserved Bandwidth for RPR in Steering Mode^{*}

Wen-Fong Wang, Yi-Shian Chen, and Lih-Chyau Wu

Department of Computer and Communication Engineering, National Yunlin University of Science and Technology, 123, Sec. 3, University Rd., Douliu, Yunlin, Taiwan
{wwf, g9517715, wuu lc}@yuntech.edu.tw

Abstract. In the RPR steering mechanism, the head stations, counted from the breakpoint on each ringlet, may seize more bandwidth of high priority traffic than their downstream nodes. This study proposes an adaptive control scheme of reserved bandwidth to throttle the add traffic flows of the head nodes and reallocate network bandwidth more fairly. With this scheme, we can control the classA1 and classB reserve rate dynamically, and adjust the downstream rate according to the congestion condition. In our simulations, we see that the classA frame delay can be decreased largely, and the global system throughput can be increased significantly.

Keywords: RPR, Resilient Packet Ring, 802.17, Protection, Steering.

1 Introduction

Resilient Packet Ring (RPR, IEEE 802.17) is a new standard for metropolitan and wide area packet networks based on dual ring topology [1]. RPR networks have multiple advantages such as spatial reuse, resilience, scalability, bandwidth efficiency, and ease of management [2].

Two protection mechanisms are employed in RPR: steering and wrapping. The steering mechanism gives better bandwidth utilization than the wrapping mechanism after a failure occurred. However, in the steering mechanism, the head stations on each ringlet may seize more bandwidth of high priority traffic than their downstream nodes. There is no research on analyzing and improving the frame delay and system throughput under the steering mode as a link failure occurs.

2 Traffic Shapers and the Protection Mechanism of RPR

In [3], RPR defines three-level service priorities according to the demand of ingress traffic. The ClassA traffic is low-latency and low-jitter and can be further divided into classes A0 and A1. The classB traffic has predictable latency and jitters and can be divided into classB-CIR (committed information rate) and B-EIR (excess information rate). Finally, the classC traffic is best effort and has unbounded jitter.

^{*} This work was supported by the National Science Council, Taiwan, R.O.C., under contracts NSC96-2221-E-224-006.

All node traffic entering the ring is throttled by rate controllers to maintain service guarantees. These controllers consist of leaky bucket shapers, which utilize credits to produce the send indications to allow the traffic of MAC clients or control messages to be transmitted [3]. The transmission rate of the three service classes is controlled by the shapers in each node. The shapers for one ringlet of a RPR node are ClassA0 shaper (shA0), classA1 shaper (shA1), classB shaper (shB), and fairness eligible shaper (shF). All of them are used to limit the add traffic of each local node for the respective traffic classes. In addition, there is a shaper for all transmit traffic other than the shapers for the add traffic, called the downstream shaper (shD).

The steering mechanism is implemented and combined with the scheme of topology database refreshment. When a link or node failure occurs, the neighboring nodes that discover the error broadcast the topology update messages on the ring to inform other nodes that the ring is broken [4]. When each node receives such a message indicating the failure, it updates its topology database accordingly. Then, steering nodes direct protected frames to the ringlet that still has connectivity to the destination of the frames.

3 Adaptive Control Scheme of Reserved Bandwidth

To resolve the unfair utilization of the classA reserved bandwidth of RPR in the steering mode, it seems that the network bandwidth reallocation by borrowing from the bandwidth of lower level classB and classC can be considered. To allocate more reserved rate to classA1, we need to dynamically adjust the credits of ClassA1 and ClassB according to the congestion situation. Therefore, we adjust the credit for classA1 according to the traffic transmission of classA1. We use two parameters to judge the range of the adjusted rate, one is *classAAccessDelayTimer* and the other is *classAAccessDelayTimerThreshold*. To measure the amount of the classA frame delay, we define a parameter *A1_weight* and assign its value as shown in (1):

$$A1\_weight = classAAccessDelayTimer / classAAccessDelayTimerThreshold \quad (1)$$

The value *hiLimitA1* is the threshold of the classA1 credit, and thus, we can adjust the classA1 credit through this value. For example, assume that the reserved_ratioA0 = 2, reserved_ratioA1 = 18, and reserved_ratioB = 36. Under these settings, the total bandwidth of the reserved rate is 100 so that the adjustable range of reserved_ratioA1 is 1~2. Therefore, we can control the value of *hiLimitA1* from 1 to 2 dynamically. According to the value of *A1_weight*, we redefine the formula for *hiLimitA1* as follows:

```

00  if A1_weight ≥ 1
01      let New_hiLimitA1 = mtuSize+ addRateA1*MAX_JITTER/2*2
02  Else
03      let New_hiLimitA1 =
04          mtuSize+addRateA1*MAX_JITTER/2*(1+A1_weight)

```

Then, we can set the value *creditA1* as shown in (2):

$$creditA1 = Min(New\_hiLimitA1, creditA1+addRateA1 * (currentTime - tickTime)) \quad (2)$$

According to the increase of the parameter reserved_ratioA1 in (2), we have to decrease the rate of reserved_ratioB to balance the borrowed bandwidth allocation of

the whole ring. We define a parameter named B_weight , and let it equal to $A1\_weight$. Then, we set the new value of $New\_hiLimitB$ in the same way as $New\_hiLimitA1$. The formula is as follows:

```

00 if  $B\_weight \geq 1$ 
01   let  $New\_hiLimitB =$ 
02      $mtuSize + (addRateA1 + addRateB) * MAX\_JITTER / 2 / 2$ 
03 Else
04   let  $New\_hiLimitB = mtuSize + (addRateA1 + addRateB)$ 
05      $* MAX\_JITTER / 2 / (1 + B\_weight)$ 

```

Then, we can set the value $creditB$ as:

$$creditB = \text{Min}(New\_hiLimitB, creditB + addRateB * (currentTime - tickTime)) \quad (3)$$

Since shD also controls the downstream traffic, we can use it to throttle the add traffic of upstream nodes together with the shA1 and shB. We define a parameter named $Down\_weight$, and let it be equal to $A1\_weight$. Then, we set the new value of $New\_hiLimitDown$ in the opposite way of $New\_hiLimitA1$. The formula is as follows:

```

00 if  $Down\_weight \geq 1$ 
01   let  $New\_hiLimitDown = hiLimitD$ 
02 else
03   let  $New\_hiLimitDown = hiLimitD / (2 - Down\_weight)$ 

```

Then, we can set the value $creditD$ as follows:

$$creditD = \text{Min}(New\_hiLimitDown, creditD + addRateD * (currentTime - tickTime)) \quad (4)$$

4 Simulation Experiments

In this section, we compare the results of the frame delay and the node throughput of classA in the normal steering mechanism and the mechanism we modified. We named it as “Adaptive Control Scheme of Reserved Bandwidth” (ACS_RB) mechanism.

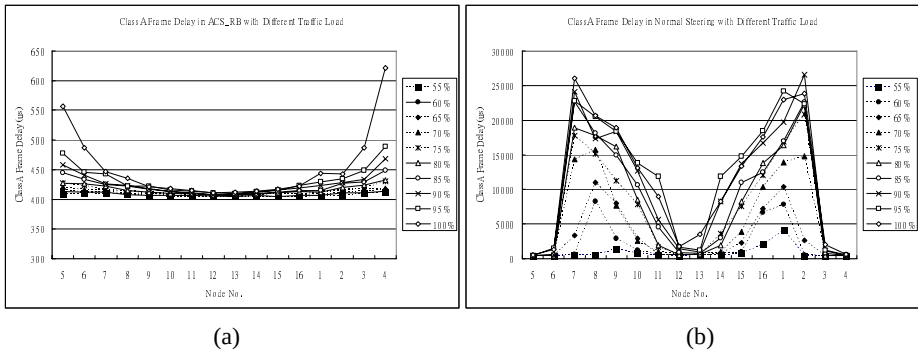


Fig. 1. Average frame delay (a) ACS_RB, (b) Normal steering condition

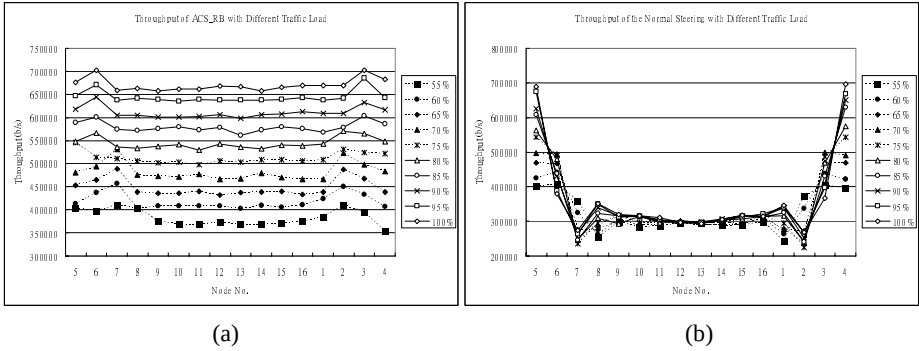


Fig. 2. Average node traffic throughput (a) ACS_RB, (b) Normal steering condition

In Fig. 1(a), we find that our proposed mechanism can make shapers shA, shB, and shD reallocate the network bandwidth more fairly. Our scheme can avoid upstream nodes to seize too much bandwidth; therefore, the downstream node may have the chance to get enough bandwidth, and its classA frame delay can be reduced greatly in comparison with Fig. 1(b).

In our scheme, we adjust the credit of three shapers dynamically, so that the classA1 traffic of the downstream node can be sent with lower delay; therefore, the whole ring throughput can be increased (see Fig. 2(a) in comparison with Fig. 2(b)).

5 Conclusion

We have proposed an enhanced scheme named ACS_RB to improve the system throughput and decrease the classA frame delay. It makes the shA1, shB, and shD control bandwidth dynamically to achieve fairer bandwidth usage in steering mode.

References

- [1] Kvalbein, A., Gjessing, S.: Analysis and improved performance of RPR protection. In: Proceeding of ICON 2004, vol. 1, pp. 119–124 (2004)
- [2] Zhu, J., Matrawy, A., Lambadaris, I., Ashourian, M.: Optimization of Resilient Packet Ring Networks Scheduling for MPEG-4 Video Streaming. In: Proceeding of ICC 2005, vol. 1, pp. 271–275 (2005)
- [3] IEEE standard for information technology-telecommunications and information exchange between systems-local and metropolitan area networks-specific requirements-part 17: resilient packet ring (RPR) access method and physical layer specifications. IEEE Std. 802.17, pp. 1–644 (2004)
- [4] Chen, Y.S.: The Simulative Analysis of IEEE 802.17 Resilient Packet Ring in QoS Properties. Institute of Computer Science and Information Engineering, National Yunlin University of Science and Technology, Master thesis (2008)

Estimating Half-Path RTT in Backbone Network

Lisheng Huang, Wenyong Wang, and Mingtian Zhou

School of Computer Science and Engineering

UEST of China

610054 Chengdu, China

Tel.: 86-28-83202180; Fax: 86-28-83202180

lsh@uestc.edu.cn

{lsh,wangwy,mtzhou}@uestc.edu.cn

Abstract. Half-path Round-Trip Time (*HPRTT*) is defined as the RTT metric from a host to the passive measurement point (MP), and a general passive method is proposed for continually measuring of HPRTT for different hosts in backbone network. Comparing with current passive RTT estimation methods, this method can obtain more measurement samples from different TCP activity scenes. Test result shows it works properly and effectively with Gbps traffic.

Keywords: Half-Path, RTT, Passive estimation, Backbone network.

1 Introduction

RTT is a time-varying uncertainty. It includes fixed delay (D_{fix}) and variable delay (D_{var}). D_{fix} is a rather stable factor which consists promulgate delay (D_p) and transfer delay (D_{tran}). D_p is decided by physical link and D_{tran} is decided by packet size and path capacity. D_{var} is variational and is infected by the host response delay and the congestion of a network path. RTT can be described as

$$\text{RTT} = D_{\text{fix}} + D_{\text{var}} = D_p + D_{\text{tran}} + D_{\text{var}} \quad (1)$$

And the HPRTT metric, as we described above, can be described as

$$\text{HPRTT} = \text{HPD}_{\text{fix}} + \text{HPD}_{\text{queue}} + D_{\text{host}} \quad (2)$$

Where HPD_{fix} is the half-path D_{fix} factor from a host to the MP. $\text{HPD}_{\text{queue}}$ stand for the queue delay cause by network congestion, D_{host} stand for the response delay time of a host(generally cause by delayed TCP ACK and data preparing).

There exists many methods for RTT estimating: SYN-ACK(SA)[1] method estimates static RTT by observing the SYN / SACK / ACK packets in a TCP 3-way handshake period; Slow-Start(SS)[1-3] method estimates RTT during TCP slow-start period by observing the RG(Round Gap) of neighboring packet pairs; Timestamp(TS)[2] method calculate RTT samples on the timestamp option which is likely to be used in a TCP conversation; Karn[4] method get RTT samples basing on the judgment of triggering relationship between DATA packet and corresponding

ACK packet, and drops suspicious samples caused by TCP Retransmission; Running[4] method maintain a finite state machine(FSM) to trace the congestion window (cwnd) of TCP sender, decide which data packet transmissions are triggered by the arrival of a particular ACK, and then finish the estimation of RTT.

In this paper we investigate several TCP activities and analyze how current existing methods can be employed and improved to obtain as many as possible valid samples.

2 Measurement Method of HPRTT

Here we investigate the factors of (2): in broadband network, the HPD_{fix} is decided by path capability. The HPD_{queue} stand for the dynamic congestion status of network path. D_{host} is the only factor which we should eliminate. We define the D_{host} factor as

$$D_{host} = D_{tcp} + D_{app} \quad (3)$$

Where D_{tcp} is the delay of TCP stack. D_{app} is the delay caused by application. Let $HPRTT_h$ be the HPRTT from host h to the MP. We have scenes are as Fig 1.

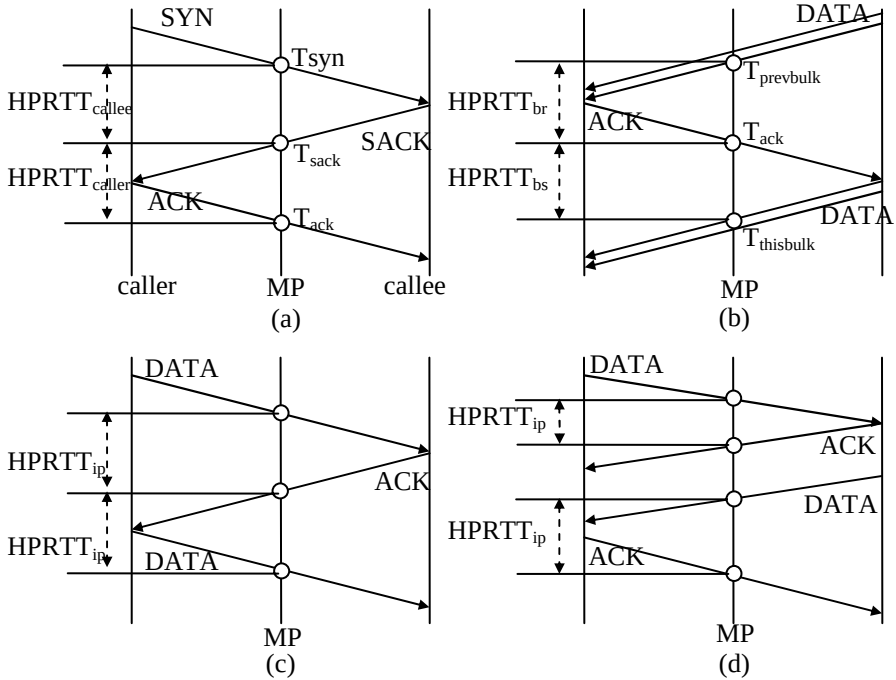


Fig. 1. Four different measurement scenes which produce different kinds of HPRTT samples

We see host h may act as different TCP roles

- Caller, host may request a TCP conversation to remote server.
- Callee, host may accept incoming TCP connection requests.

- Bulk sender / receiver, host may send / receive bulk data in a TCP conversation.
- Interactive participant, host may Participate in a interactive session, such as telnet.

Thus we get several types of samples, $HPRTT_{caller}$ for the caller, $HPRTT_{callee}$ for the callee, $HPRTT_{bs}$ for the Bulk sender, $HPRTT_{br}$ for the Bulk receiver, and $HPRTT_{ip}$ for the interactive participant.

In (a) we can get $HPRTT_{caller}$ and $HPRTT_{callee}$ samples with SA-like method, in (b) we can get $HPRTT_{br}$ and $HPRTT_{bs}$ with SS-like method, in (c) and (d) we can get $HPRTT_{ip}$ with Karn-like or Running-like method.

We maintain a minimize $HPRTT$ sample called $HPRTT_{min}$ for each measured host, which can be used as a standard as deciding whether a sample is valid or not. Samples similar to $HPRTT_{min}$ will be treated as valid ones, and other samples may be doubtful and need more analyzing and judgement. $HPRTT_{min}$ is considered not impacted by D_{host} and HPD_{queue} , so according to (2), $HPRTT_{min} \approx HPD_{fix}$, at the beginning of measurement period, let $HPRTT_{min}$ to be the first $HPRTT$ sample and it is replaced by any new samples with a less value.

We also designed a self-adaptive *gating* filter, to judge the validity of measurement samples and wipe off measurement noise. The basic idea is to define a self-adaptive threshold value, and then compare each measurement sample with this threshold.

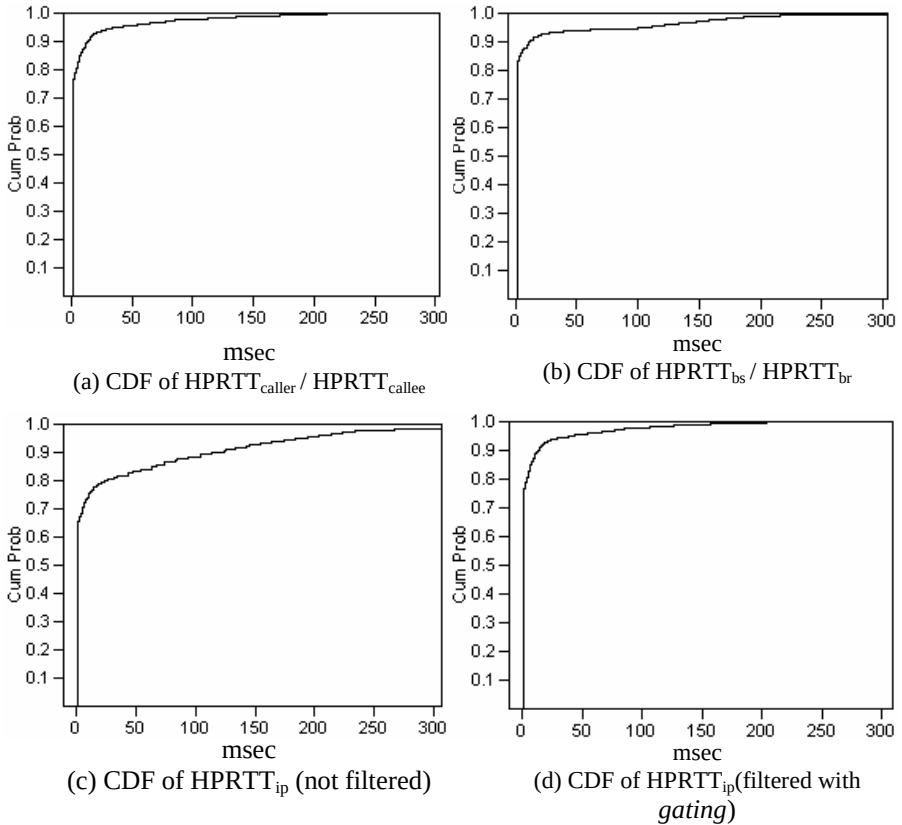


Fig. 2. Cumulative distribution of different $HPRTT$ samples

3 Measuring HPRTT in Backbone Network

We deployed a passive MP at CERNET south-west center, TCP traffic between local network and CERNET are analyzed.

We implemented our method into a software named RPPM, which runs on a PC server with two 2.4GHz CPU and windows xp system. We captured Gbps level traffic, 32,240,581 TCP conversations are investigated, including 2374 local hosts, 1094 CERNET hosts, and 136,353 internet hosts. From everyday's PING test, we've got to know all RTTs from local hosts to MP are less than 30ms. After hours of continual measurement, RPPM itself did not encounter packet loss with local NIC.

To verify the validity and veracity, we saved all HPRTT samples for local hosts and classify them as different types, and their cumulative distributions are as fig 2.

We see more than 90% percent of HPRTTcallee/HPRTTbs/HPRTTbr samples are very close to real network delay (as we have already known), this proves the availability of our method. We also noticed most HPRTTcaller samples are also valid and even do not need a gating filter - this may be caused by the simplification of local hosts' operation system. As we have analyzed, a considerable percent of the HPRTTtip samples contain noise samples, the gating filter model works well at this time and the CDFs before and after filtering are as (c) and (d).

4 Conclusion

We propose HPRTT as a half-path RTT metric which can be facilitated to validate the service level for end users, we believe this metric is more useful in daily network management. On the basis of current existing passive RTT estimation methods, our new method can accurately retrieve measurement samples from most TCP activity scenes, and our new *gating* filter model is proved available to eliminate measurement noise.

References

1. Hao, J., Dovrolis, C.: Passive estimation of TCP round-trip times. ACM SIGCOMM Computer Communication Review 32(3), 75–88 (2002)
2. Veal, B., Li, K., Lowenthal, D.: New Methods for Passive Estimation of TCP Round-Trip Times. In: Proceedings of Passive and Active Measurements (PAM), pp. 121–134 (2005)
3. Zhang, Y.B., Lei, Z.: A Passive RTT Estimate Algorithm for TCP. Journal of Beijing University of Posts and Telecommunications 27(5), 85–89 (2004)
4. Jaiswal, S., et al.: Inferring TCP connection characteristics through passive measurements. In: IEEE INFOCOM (2003)
5. Karn, P., Partridge, C.: Improving Round-Trip Time Estimates in Reliable Transport Protocols. In: ACM SIGCOMM, August 2-7 (1987)

A Radio Network Co-design System for Planning, Operation, and Customer Relations Divisions

Kosei Kobayashi¹, Yasuhiko Matsunaga¹, Takayuki Nyu², and Hiroto Sugahara¹

¹ System Platforms Research Laboratories, NEC Corporation,
1753 Shimonumabe, Nakahara-ku, Kawasaki, Kanagawa, 211-8666 Japan
{k-kobayashi@ew, y-matsunaga@bl, h-sugahara@cb}.jp.nec.com

² Common Platform Software Research Laboratories, NEC Corporation,
1753 Shimonumabe, Nakahara-ku, Kawasaki, Kanagawa, 211-8666 Japan
t_new@az.jp.nec.com

Abstract. Radio network planning after the initial rollout phase has a multitude of individual coverage and capacity problems. Such problems are often exposed by customer complaints or performance indicator degradations, so cooperation among network planning, operation, and customer relations divisions will help to more effectively improve network quality. In this paper, we propose to introduce radio network co-design by multiple divisions for sharing their issues and discussing possible solutions in the same place. To realize network co-design, we developed methods of 3D visualization of radio quality for enhancing comprehension of geographically dependent radio problems, high-speed prediction of radio propagation for interactive network planning, and estimation of radio resource usage for mutual understanding of the outcome of radio planning. Finally, we demonstrate an interactive wireless network design prototype system based on these methods.

Keywords: Radio Network Planning, Performance Management, Co-design.

1 Introduction

Most cellular radio network operators have finished the initial network rollout for expanding area coverage and are now in the popularization phase. In accordance with this change, the focus of quality-improvement operations for radio networks is shifting from wide-area radio coverage expansion to countermeasures against a multitude of individual network problems. We expect that improving network quality will become more difficult for a network planning division working alone because such individual problems are only exposed by customer complaints or performance indicator degradations. Thus, radio planning by only the planning division may produce inappropriate countermeasures that ignore the seriousness of customer complaints or cause unanticipated traffic congestion.

To address these issues, we propose to introduce radio network co-design by the planning, operation, and customer relations divisions for fast and effective network quality improvement in the network popularization phase by enabling them to share their issues and discuss possible solutions in the same place.

2 Radio Network Co-design

In the initial network rollout phase, the aim for radio planning is to achieve sufficient radio coverage by installing a minimum number of base stations [1]. At this point, the planning division can design a radio network independently. In the network popularization phase, on the other hand, the number of subscribers and amount of network traffic increases, resulting in a multitude of individual network problems. Examples of such problems are insufficient indoor radio coverage, serious interference in high-rise buildings, and time-dependent resource shortages in localized areas, which are hard to find by field drive tests on main streets. These problems are often exposed by end-user complaints received by the customer relations division [2] or the degradation of key performance indicators (KPIs) monitored by the operation division [3]. Therefore, it is important that multiple divisions participate in radio planning work and discuss the possible ways to improve network quality.

We therefore propose to introduce radio network co-design by multiple divisions for fast and effective network quality improvement in the network popularization phase. A conceptual diagram of a network co-design environment is shown in Fig. 1. The planning division inputs area planning and field measurement data and obtains return-on-investment estimations of radio quality resulting from expansion of the network equipment such as base stations. The operation division inputs time-dependent KPI data and obtains an estimation of KPI improvement due to countermeasures such as equipment expansion and antenna tuning implemented by the planning division. The customer relations division inputs customer complaints data and can confirm that the problems are resolved by the countermeasures implemented by the planning division.

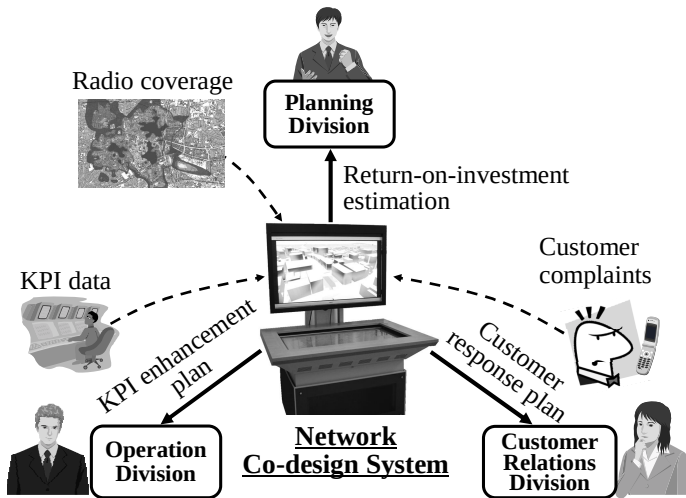


Fig. 1. Conceptual diagram of network co-design environment

3 Methods to Support the Network Co-design

To achieve network quality improvement through cooperation between multiple divisions, we developed the following three methods.

(a) 3D Visualization of Radio Quality

Participants from multiple divisions have different work responsibilities. Therefore, the system must be able to visualize radio quality information to the participants who are not familiar with complex radio propagation in urban areas. For this problem to be resolved, we took the approach of visualizing radio quality on a 3D map in combination with a 2D map, in which the spatial distribution of radio quality is represented by coloring building surfaces according to radio quality. By using this method, all participants can visually confirm the spatial distribution of radio quality.

(b) High-speed Radio Propagation Prediction

Participants from multiple divisions will discuss various countermeasures against network quality problems. Therefore, the system must quickly respond and predict radio propagation not to interrupt their ongoing discussion. To achieve this, we developed high-speed prediction methods by applying both ray-launching and statistical method for fast estimation of diffracted components, and by object combining to reduce the number of objects used in an analysis [4]. By using these methods, we can estimate the radio quality in an urban area within several tens of seconds per sector, and therefore co-design participants can evaluate the various countermeasures without interrupting their ongoing discussion.

(c) Estimation of Radio Resource Usage

In the network popularization phase, not only radio quality but also radio resource usage must be confirmed to prevent unanticipated traffic congestion. For the resolution of this issue, we developed a method that estimates future radio resource usage while accounting for the change of radio environment caused by a countermeasure. The method first estimates radio resource usage per unit area based on the dominant cell coverage and past radio resource usage. Then, we predict future radio resource usage by allocating above radio resource usage per unit area to the dominant cell coverage after radio planning obtained by radio propagation prediction. By using this method, the planning division can design a radio network with sufficient capacity by considering the traffic characteristics of the target area.

4 Prototype System

We developed an interactive wireless network design prototype to prove the viability of the network co-design concepts. It consists of two large screens at right angles. One is a horizontal table display, which is for visualizing wide-area network quality and shows base stations, customer complaints, and radio quality overlaid on a 2D map, as shown in Fig. 2(a). The other is a vertical display, which is for detailed and supplementary information on the target area and shows radio network information overlaid on a 3D map and an operational KPI data graph, as shown in Fig. 2(b).

This system enables us to grasp the radio quality in a wide area by the 2D map and to evaluate detailed radio quality near a complaint area by the 3D map. Analysis results after adding a base station or tuning an antenna can be obtained within 20 to 30 seconds, which is fast enough for interactive network design. Furthermore, this system can promote cooperation between multiple divisions by representing various information, such as radio quality, operational KPI, and customer complaints, on the two displays.

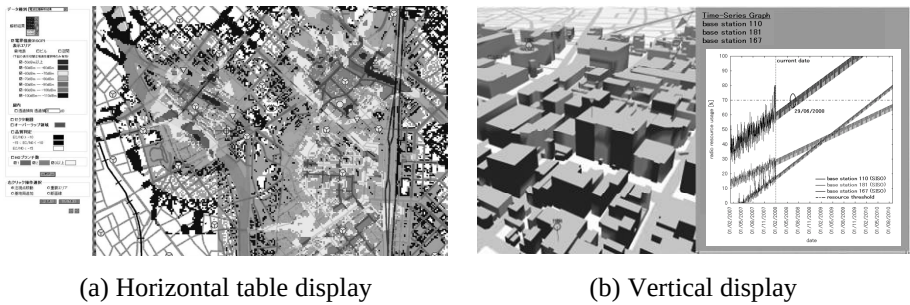


Fig. 2. Screenshots of interactive wireless network design prototype

5 Conclusion

We proposed radio network co-design between planning, operation, and customer relations divisions for sharing their issues and discussing possible solutions in the same place. To achieve network co-design, we developed methods for the 3D visualization of radio quality, high-speed prediction of radio propagation, and estimation of radio resource usage. We demonstrated the feasibility of the network co-design concepts with a prototype system that enables interactive network planning and supports mutual understanding among multiple divisions.

References

1. Laiho, J., Wacker, A., Novosad, T.: Radio Network Planning and Optimisation for UMTS. John Wiley & Sons, Ltd., Chichester (2002)
2. Rosenberg, A., Kemp, S.: CDMA Capacity and Quality Optimization. McGraw-Hill, New York (2003)
3. Watanabe, Y., Matsunaga, Y., Kobayashi, K., Tonouchi, T., Igakura, T., Nakadai, S., Kamachi, K.: UTRAN O&M Support System with Statistical Fault Identification and Customizable Rule Sets. In: 11th IEEE/IFIP Network Operations and Management Symposium, Salvador, Brazil, AS 3, pp. 560–573 (2008)
4. Motoyoshi, M., Sugahara, H., Matsuda, Y., Ono, T.: High-Speed Estimation of MIMO System Capacity using a Hierarchical Approach and Preprocessing Digital Maps. In: 2008 IEEE International Symposium on Antennas and Propagation, San Diego (2008)

Traffic Matrix Estimation Using Square Root Filtering/Smoothing Algorithm

Jingjing Zhou¹, Jiahai Yang², Yang Yang³, and Guanqun Zhang⁴

^{1,3}Information Engineering School, University of Science and Technology Beijing
Beijing, China, PA 100083

^{1,2,4}The Network Research Center, Tsinghua University,

^{1,2,4}Tsinghua National Laboratory for Information Science and Technology,
Beijing, China, PA 100084

zhoujingjings@gmail.com, yang@cernet.edu.cn, yyang@ustb.edu.cn,
zhangguanqun@gmail.com

Abstract. The traffic matrix (TM) is one of the crucial inputs in many network management and traffic engineering tasks. As it is usually impossible to directly measure traffic matrices, it becomes an important research topic to infer traffic matrix by reasonably modeling, and incorporating the measurement data of limited links, as well as other additional information. In this paper, we propose Square Root Filtering/Smoothing traffic matrix estimation (SRFsTME) algorithm based on Kalman Smoothing decomposition to improve our proposed Square Root Kalman Filtering traffic matrix estimation (SRKFTME) algorithm. Simulation and actual traffic testing results show that SRFsTME algorithm is more numerical accurate and stable than the SRKFTME algorithm.

1 Introduction

The origin-destination (OD) traffic matrix (TM) is critical input for network configuration and management, traffic engineering etc. It describes the amount of traffic that flows between every possible pairs of ingress and egress points in a network. However, direct measurements to obtain traffic matrices are prohibitively expensive[1], so many estimation methods based on limited available information are used to obtain the traffic matrices. The important problem of the traffic matrix estimation is the exist of the ill-posed linear inverse problem[2]. Many of the previous work on IP traffic matrix estimation have incorporated prior information to solve the ill-posed linear inverse problem[2, 3, 4]. However these methods have not been able to push error rates sufficiently low for carriers to use the estimated traffic matrix. So, there are other estimated methods[5, 6] appearing recently.

Of the proposed methods, Kalman Filtering method is a more efficient and accurate method than many others [7]. However, we find that the error covariance calculation components of the Kalman Filtering method are difficult to implement in realistic systems due to the existence of the ill-conditioning problems. We have proposed Square Root Kalman Filtering traffic matrix estimation (SRKFTME) algorithm [8] based on matrix decomposition to improve the Kalman Filtering method. The

SRKFTME algorithm is an important improvement of the standard Kalman Filtering method, but the performance can not reach our goals. In this paper, we propose Square Root Filtering/Smoothing traffic matrix estimation (SRFsTME) algorithm based on the Kalman Smoothing decomposition to improve the SRKFTEM algorithm. Simulation and actual traffic testing results show that the performance of the SRFsTME algorithm is superior to the SRKFTME algorithm.

This paper is organized as follows. Section 2 will briefly introduce the Kalman Filtering method proposed in[7]. In Section 3, we propose the SRFsTME algorithm to improve Kalman Filtering method based on the proposed SRKFTME algorithm [8]. In Section 4, we test the performance of the SRFsTME algorithm comparing with the SRKFTME algorithm, and present the concluding remarks.

2 The Kalman Filtering Method

The Kalman Filtering traffic matrix estimation method[7] uses state space models from dynamic linear system theory to capture the evolution of the network systems. Let the vector x_t denote the entire set of OD traffic at discrete time t . Let y_t denote the set of all observations up to (and including) time t . So, the evolution of the traffic states and the observation equation can be modeled as the following linear system,

$$x_{t+1} = Cx_t + w_t \quad (1)$$

$$y_t = Ax_t + m_t \quad (2)$$

In this model C and A is the state transition matrix and routing matrix respectively. w_t and m_t is the traffic system and measurement process noise respectively.

Let $\hat{x}_{t|t}$ denote the *estimation* of x_t at time t . Let $P_{t|t} = E \left[\left(\hat{x}_{t|t} - x_t \right) \left(\hat{x}_{t|t} - x_t \right)^T \right]$ denote

the covariance matrix of estimation errors at time t . Both the state-noise w and the measurement-noise m are assumed to be zero-mean white-noise processes, uncorrelated and with covariance matrices Q and R , respectively. Let the initial conditions of system states be denoted by $\hat{x}_{0|0} = E[x_0]$ and $P_{0|0}$. To produce the estimation in equations

(1) and (2), $\hat{x}_{0|0}$ and $P_{0|0}$ need to be known, and the matrices C , Q and R need to be estimated using the EM algorithm[7].

3 The SRFsTME Algorithm

When using the Kalman Filtering method to estimate the traffic matrix of the CER-NET2 backbone, which is the second generation academic network in China, we find that the error covariance calculation components of the Kalman Filtering are difficult to implement in realistic network systems, the numerical difficulties have occurred in problems with singular and nearly singular estimation covariance matrices because of the ill-conditioning in the MATLAB environments [8]. To cope with these ill-conditioning situations of Kalman Filtering, we designed the SRKFTME algorithm

to improve it[8]. But, the performance can not reach our goals. In this paper, we propose SRFsTME algorithm to improve the SRKFTME algorithm. The SRFsTME algorithm is implemented in a form of feedback control through the two filtering steps of SRKFTME algorithm and one smoothing step as follows:

(1) Filtering Prediction Step: We use the MWGS (Modified Weighted Gram-Schmidt Orthogonalization) [9] to implement the traffic matrix prediction to replace that of the standard Kalman Filtering. It is an alternative algorithm with better overall numerical stability and accuracy [8].

(2) Filtering Estimation Step: This step updates the state and variance using a combination of the predicted state and the observation [8]. This estimation step continues with the factorization used in the smoothing updating step such that we can use the obtained UD matrices to obtain an estimate of $P(+)$.

(3) Smoothing updating step: We use the UD covariance factorization of Rank-2 Smoother algorithm[10] to implement the smoothing updating step for realizing the EM algorithm to estimate the parameters C , Q and R of the filtering equation. The rank-2 result, equation (4), turns out to have all positive attributes that are sought. The Rank-2 Smoother algorithm is:

$$x_{t/T}^{(w_j)} = x_{t+1/T}^{(w_j)} + g_t \delta_t \quad (3)$$

$$P_{t/T}^{(w_j)} = (I - \lambda_t g_t v_t^T) P_{t+1/T}^{(w_j)} (I - \lambda_t g_t v_t^T)^T + \lambda_t g_t g_t^T \quad (4)$$

Where $G = [g_1, \dots, g_T]$, $Q = \text{diag}(q_1, \dots, q_T)$, $v_t = \left[\hat{P}_t^{(w_j)} \right]^{-1} g_t$, $\lambda_t = \frac{q_t}{1 + q_t v_t^T g_t}$,

$P_{t+1/T}^{(w_j)} = P_{T/T}^{(w_j)} (+)$, $x_{t+1/T}^{(w_j)} = x_{T/T}^{(w_j)}$ the superscript notation w_j employed here is intended for clarity of exposition. To be specific, let the matrix-related vectors be compatibly partitioned and computed as follows [10]:

$$U_{t+1/T} = \begin{bmatrix} U_{t+1/T}^{(1)} & U_{t+1/T}^{(2)} \\ 0 & U_{t+1/T}^{(3)} \end{bmatrix}; \quad D_{t+1/T} = \text{Diag}(D_{t+1/T}^{(1)}, D_{t+1/T}^{(2)}); \quad (5)$$

$$v^T = (v_1^T \quad v_2^T) = v_t^T U_{t+1/T}; \quad b_t^T = (c_1^T, 0)$$

$$\begin{aligned} U_{t/T}^{(1)} D_{t/T}^{(1)} (U_{t/T}^{(1)})^T &= (I - \lambda_t c_1 c_1^T) U_{t+1/T}^{(1)} D_{t+1/T}^{(1)} (U_{t+1/T}^{(1)})^T + \lambda_t c_1 c_1^T; \\ U_{t/T}^{(2)} &= U_{t+1/T}^{(2)} - \lambda_t c_1 f_2^T; \quad U_{t/T}^{(3)} = U_{t+1/T}^{(3)}; \quad D_{t/T}^{(2)} = D_{t+1/T}^{(2)} \end{aligned} \quad (6)$$

When the equation (3) to (4) is repeated for $t=T_w, T_w-1, \dots, 1$, $x_{t/T}^{(w_j)}$ and $P_{t/T}^{(w_j)}$ will be reached respectively. The results $x_{t/T}^{(w_j)}$ and $P_{t/T}^{(w_j)}$, $t=T, T-1, \dots, 1$, are important inputs for the *expectation step* of EM algorithm to calibrate the parameters C , Q and R .

The above three steps are repeated until the system parameters of the *maximization step* of EM algorithm are converged. When the iterations are terminated, the output of the filtering prediction step, x_t , $t=T, T-1, \dots, 1$, is our estimated traffic matrix.

4 Algorithm Validations and Conclusions

We estimated the POPs level traffic matrix based on the measurable data of the CERNET2 backbone to evaluate our SRFsTME algorithm. This grained traffic matrices are generally fine enough to analyze the situations of CERNET2 backbone for us. In our experiment, we compare the OD TM estimated by standard SRKFTME and SRFsTME algorithm with the original OD traffic obtained through router port mirroring method. From the results, we find that the estimated TM using SRFsTME algorithm fit well with the original OD traffic at most points, and the accuracy and stability have a great improvement comparing with the SRKFTME algorithm.

In this paper, we propose Square Root Filtering/Smoothing traffic matrix estimation (SRFsTME) algorithm based on Kalman Smoothing decomposition to improve the SRKFTME algorithm. Simulation and actual traffic testing results show that the performance of the SRFsTME algorithm is superior to the SRKFTME algorithm. In our future work, we will compare the computation rates and error rates of the SRFsTME algorithm with other estimation methods.

Acknowledgements. This research work is supported by the National Natural Science Foundation of China under Grant Nos.90412012, 60673160, 60473083, and the National High-Tech Research and Development Plan of China under Grant No.2007AA01Z234, 2006AA01Z209, 2007AA01Z2A2, 2003CB314807.

References

1. Konstantina, P., Nina, T., Anukool, L.: A Distributed Approach to Measure Traffic Matrices. In: IMC 2004. ACM Taormina, Italy (2004)
2. Vardi, Y.: Network tomography: Estimating source-destination traffic intensities from link data. *Journal of the American Statistical Association* 91, 365–377 (1996)
3. Yin, Z., Matthew, R., Carsten, L., David, D.: An information-theoretic approach to traffic matrix estimation. In: SIGCOMM 2003. ACM Karlsruhe, Germany (August 2003)
4. Alberto, M., Nina, T., Kave, S., Supratik, B., Christophe, D.: Traffic matrix estimation: Existing techniques and new directions. In: SIGCOMM 2002, pp. 161–174 (August 2002)
5. Antonio, N., Rene, C., Nina, T., Christophe, D.: Design of IGP Link Weight Changes for Estimation of Traffic Matrices. In: Infocom 2004. IEEE, Hong Kong (2004)
6. Gang, L., Nina, T., Bin, Y.: A Fast Lightweight Approach to Origin - Destination IP Traffic Estimation Using Partial Measurements. *IEEE Transactions on Information Theory* 52 (June 2006)
7. Augustin, S., Kave, S., Antonio, N., Nina, T.: Traffic Matrix tracking using Kalman Filtering. In: Taft-LSNI 2005 (2005)
8. Jingjing, Z., Jiahai, Y., Yang, Y., Hongbo, L.: Traffic Matrix Estimation Based on Square Root Kalman Filtering Algorithm. *International Journal of Network Management* 18(5) (2008)
9. Bierman, G.J.: *Factorization Methods for Discrete Sequential Estimation*. Academic Press, New York (1977)
10. Bierman, G.J.: A new computationally efficient fixed-interval, discrete-time smoothers. *Automatica* 19, 503–511 (1983)

A Knowledge-Based Tool to Support Clear Relationship between Threats and Countermeasures Based on International Standards

Guillermo Horacio Ramirez Caceres and Yoshimi Teshigawara

Graduate School of Engineering, Soka University
1-236 Tangi-cho, Hachioji, Tokyo 192-8577, Japan
{guillerm,teshiga}@soka.ac.jp

Abstract. In this paper, we propose a web application for security policy management by using threats-countermeasure model based on international standards. The main objective of this research is to create a knowledge-base to identify and specify the threats that affect the IT environment. In addition, our proposed knowledge-base system intends to fuse similar security controls or objectives to create effectively a security guideline for a specific IT environment. This web application display a detail of the most common threats to the information systems, and for each threat, present a set of related security controls from different international standards including ISO/IEC 27002.

1 Introduction

Many international standards exist in IT security field. This research is based on ISO/IEC 15408, 15446, 13335 and 17799 [1]-[4]. The security information can be understood as the ability of an information system which uses the Evaluation Assurance Levels (EAL) as defined in international standards ISO/IEC 15408 to avoid all accidents or malicious deliberate actions. ISO/IEC 27001, also known as Information Security Management System (ISMS), is an international standard to be used as a guideline for initiating, implementing, maintaining, and improving information security management in an organization [5]. ISO/IEC 27002 provides implementation guidance in order to implement security controls. Therefore, large amount of times, cost, and knowledge are required to implement a security system. The issues described above motivated us to develop this knowledge-based tool to support users in order to implement a security system. This research includes a simple application, which is easily used for the common users, to evaluate the threats affecting their system environments.

This paper proposes a threats-countermeasures model to be used to identify and specify the threats that affect the IT environment. In addition, our proposed knowledge-base system, intends to fuse similar security controls or objectives to create effectively a security guideline for a specific IT environment. This security objective knowledge-base was developed by using the relationship inside the standards and the relationship to other standards.

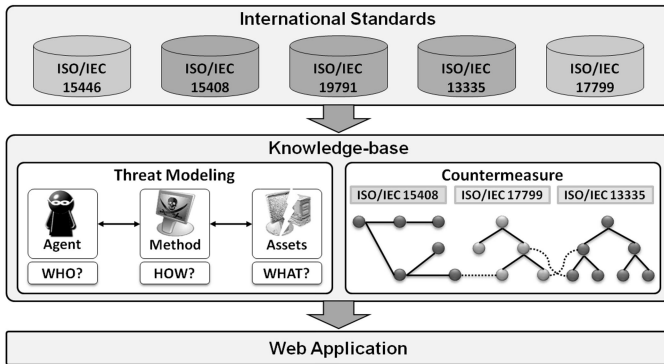


Fig. 1. Proposed knowledge-based model

2 Knowledge-Base Architecture

Our main research target is the construction of a knowledge-base system to select security policies for secure and trustable IT environment based on multiples international standards. As shows in Fig 1 the threat-countermeasure architecture is introduced. Our proposed method has been divided into two steps. The first one is to identify and specify the threats that affect the IT environment. Each of the identified threats to security is addressed by one or more security objectives. Next step is to specify the countermeasures, also called security objectives. These objectives are suitable to counter all identified threats.

2.1 Threats Information

The attacks can exploit several objectives including fraud, extortion, and robbery of information, revenge or simply the challenge to penetrate systems. This can be made by internal employees who abuse their permissions of access, or by external attackers who break in remotely or intercept the network traffic. The majority of successful attacks on operating systems link to only a few vulnerable software. They take the easiest and most convenient route, and exploit the best-known flaws with the most effective and widely available attack tools.

To identify and specify the threats that affects the IT environment, based on the security concept described before, we has been developing a threat model based on multiple international standards[6].

To identify and specify the threats is necessary to know:

- Who is the person who becomes a threat? (WHO)
- How the attack technique is used? (HOW)
- What is the object exposed to the threat?(WHAT)

2.2 Countermeasures Information

There are too many security objectives in different international standards. We have been developing a knowledge-base for security objective including: ISO/IEC 15408, 17799 and, 13335.

ISO/IEC 15408, known as Common Criteria (CC) for Information Technology Security Evaluation, is an international standard used as the basis to evaluate the security properties of IT products. In this threats-countermeasures model, we include security objectives described in part 4 of Security Target (ST) evaluated by CC.

ISO/IEC 27002 contains eleven security control clauses. Each clause contains a number of security categories. Finally, these security categories include a control objective and one or more controls. Our proposed knowledge-base includes these 11 clauses, 39 security categories, and 133 security controls. However, there are too many relationships between these security controls. For example, some security control includes a reference “see also x”, where “x” is the numerical value shown another security control to be included. However, our knowledge-base works as a Web application allows users to search easily and quickly the necessary security objectives and relates security objectives inside ISO/IEC 27002 and other standards.

In order to select countermeasures according to this threats classification, we also include in this knowledge base information of ISO/IEC 13335, in particular guidance on the selection of safeguards from Part 4 of ISO/IEC 13335.

3 Knowledge-Base Application

Our proposed knowledge-base application was created to be used for home users, who do not have the special knowledge. This tool allows to access to information

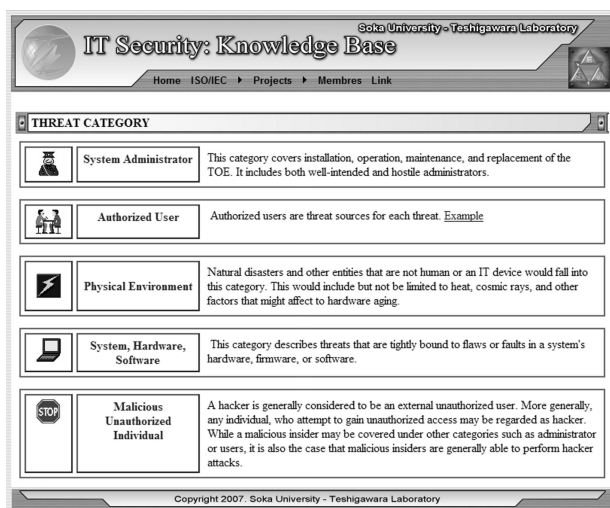


Fig. 2. Threats categories

of the threats that affect the IT environment. Users can search for some threats by selecting some parameters. Furthermore, the threats have been classified into five categories: system administrator, authorized user, physical environment, systems hardware, and malicious unauthorized individual. Fig 2, shows a snapshot of our developed web application so that a user can find the necessary information about threats quickly and effectively.

In this knowledge-base, each security threat is mapping to one or more security objectives including in international standards. After identified a threats, a user is able to read about security control information from international standards. Then, the next step is to select countermeasures or security objectives to counter risks of possible attacks.

4 Conclusion and Future Work

A security policy is an organizational tool to inform users about the importance and sensibility of the information and critical services that allow the company to grow and to stay competitive.

This threat model architecture is based on ISO/IEC 15446 and ISO/IEC TR 13335. In addition, our proposed model aids users in the process of creating security policies by selecting the appropriate security controls agilely and effectively, in accordance with the IT environment, since the user operates only with a minimal set of security objectives. Moreover, all security policies in this model are created for the respective threats.

References

1. ISO/IEC 15408. Common Criteria for Information Technology Security Evaluation Part 1-3. Version 3 CCIMB-99-031 (August 1999)
2. ISO/IEC TR 15446. Information technology - Security techniques - Guide for the production of protection profiles and security targets (2004)
3. ISO/IEC TR 13335-1-5, Information technology - Guidelines for the management of IT Security
4. ISO/IEC 17799. Information technology - Code of practice for information security management (2000)
5. ISO/IEC 27001. Information technology - Security techniques - Information security management systems - Requirements (2005)
6. Ramirez Caceres, G.H., Teshigawara, Y.: Proposal of a Creation Method for Secure and Trusted IT Environment Based on Multiple International Standards. In: APNOMS 2006, Busan, Korea (September 2006)

Incentives for Cooperative Relay in Heterogeneous Networks: A Dynamic Reputation-Based Approach^{*}

Junseok Hwang, Andrei Shin, and Hyenyoung Yoon

College of Engineering, Seoul National University,
San 56-1 Shinlim-Dong, Gwanak-Gu, Seoul, 151-742, Republic of Korea
{junhwang, andrei, hyyoon00}@snu.ac.kr

Abstract. Selfish behavior by users significantly affects the overall performance and degrades the overall connectivity in heterogeneous networks, such as Personal Networks (PN). Incentives are considered one of the best solutions for this situation. This paper proposes a dynamic reputation-based incentive mechanism for cooperative relay in PNs. Based on reputation calculations for individual contributions of intermediate relay nodes in a cooperative relay service, the proposed scheme dynamically assigns incentives in terms of additional throughput for those cooperative relay nodes. Experimental results of simulation scenarios show that dynamic incentives are appropriate and sustainable for motivating user participation in PNs, particularly for applications characterized by wide coverage and high data rates.

1 Introduction

Rapid technological development and the convergence of different technologies deliver a diversity of visions for next-generation networks. To realize a PN [1], an existing 3G cellular network base station (BS) can act as the PN gateways and the mobile operator (MO) can play the role of PN provider. The BS may not be reachable from a particular device because of a resource deficit, signal fading, shadowing or some other location-dependent factor. To provide the cooperative relay service, additional consumption of valuable resources such as radio spectrum and battery power is required. Users will not be voluntarily interested in cooperating with others' traffic transmission unless there are incentives. In this paper, we propose a dynamic reputation-based incentive mechanism for PNs that gives additional throughput as a reward for individual participation in, and contribution to, the relaying service for other network users.

2 Dynamic Reputation-Based Incentive Model

In a two-hop-relay architecture, the radio resources at the BS are allocated via a time-division high-data-rate system scheduler. However, under the assumption that all users are selfish, only the proportional fair scheduler offering additional throughput as

^{*} This work was supported by the MKE, Korea, under the Information Technology Research Center (ITRC) support program supervised by the IITA.

a fixed incentive provides positive encouragement of users' participation in the cooperative relay [2].

A dynamic incentive mechanism can be achieved via the addition of an incentive (bonus) parameter in the timeslot allocation of the proportional fair scheduler. The incentive parameter b_i is set to 1 when a user node does not participate in the cooperative relay service. For relay nodes, the incentive variable has two components: a fixed-bonus variable (b_{Fi}) and a dynamic variable (b_{Di}) that accounts dynamically for the user contribution. Timeslot allocation can therefore be expressed as:

$$k = \arg \max_{\forall i} \left\{ \frac{b_i \times r_i(t)}{\mu(t)} \right\} . \quad (1)$$

where k is the time slot, b_i is the incentive parameter, $r_i(t)$ is the maximum ratio of the instantaneous data rate, and $\mu(t)$ is the average of the former allocated rate.

$$b_i = \begin{cases} (b_{Fi} + b_{Di}) & \text{for relay nodes} \\ 1.0 & \text{for user nodes} \end{cases} . \quad (2)$$

The component b_{Fi} is a base value representing the initial percentage increase in throughput offered by service providers based on their business model for all nodes that decide to become relay nodes in the cooperative relay scheme. This initial fixed throughput reward will depend on the service provider's evaluation of the relative importance of each relay node to the system.

The component b_{Di} represents the dynamic part of the reward. Its value is assigned dynamically, being based on the available reputation for assistance in the cooperative relay. To calculate b_{Di} , we applied the SORI architecture used in [3].

$$b_{Di} = \frac{\sum_{j \in NNL_i \cup \{i\}, j \neq X} \lambda_i(j) \cdot C_j(X) \cdot G_j(X)}{\sum_{k \in NNL_i \cup \{i\}, k \neq X} \lambda_i(k) \cdot C_k(X)} . \quad (3)$$

where NNL is a Neighbor Node List for user node i , j and k are nodes in the NNL , $\lambda_i(j)$ and $\lambda_i(k)$ are the credibilities of user nodes j and k to i , $C_j(X)$ and $C_k(X)$ are the confidences of user nodes j and k about relay node X . $G_j(X)$ is $HF_j(X)/RF_j(X)$.

In this model, each end user dynamically calculates the reputation of each relay node that has contributed to traffic transmission. Each end user then periodically updates its reputation record and submits it to the BS. The BS calculates its own reputation record for each relay node from this information and uses it to give rewards in terms of appropriate additional throughput to each contributing relay node.

3 Agent-Based Modeling and Simulation Description

Agent-Based Modeling (ABM) is a computation-based approach to simulating actions and interactions between autonomous individuals in a network. In our ABM model

each involved party is represented by an agent: there are base-station agents, relay-node agents, and user-node agents.

Ultra High Frequency (UHF) is the frequency band for transmission of television signals. Nowadays, modern mobile devices can transmit and receive within the UHF spectrum band. However, UHF TV signals are sensitive to external noise and the signal strength reduces over long distances. Therefore, transmission of UHF TV signals can be improved by applying the cooperative relay service concept.

We evaluate our scenario through agent-based simulation in which the two-hop-relay network topology has one BS, four relay-node agents and eight user-node agents. In the first experiment, we analyzed the impact of our dynamic scheme on different data rates. For the second experiment, we chose only one data rate for the simulation and examined several combinations of initial weights for the base component b_{Fi} and the dynamic value b_{Di} to observe the behavior of the model.

4 Experimental Results

Figure 1a) shows the results of Experiment 1, for all data rates (38.4 Kb/s, 76.8 Kb/s, and 153.6 Kb/s) and all types of throughput allocation (normal, fixed, and dynamic). For instance, using the lowest data rate (38.4 Kb/s), the dynamic incentive mechanism shows almost the same results as the fixed reward allocation initially but, as time goes on, the dynamic incentive allocation shows better results (after 250 seconds of simulation). Until this time, the relay nodes are in the initial stage of accumulating reputation values, which gives only small rewards. Therefore, their total throughput was almost the same as for fixed throughput allocation. From 250 seconds onwards, reputation values reach a point where the increasing reputation value positively affects the total throughput. After 500 seconds of simulation, the total throughput has an almost constant shape, indicating the maximum available throughput reward in the system. For higher data rates, the dynamic incentive mechanism show superior results.

Figure 1b) shows the results of Experiment 2, where we analyzed the dynamics of incentives, using different initial values, and compared them with the fixed incentive mechanism at the lowest data rate setting.

The results showed that using dynamic incentive allocation alone, without any initial fixed incentives (shown as “Dynamic 1.0 + Reputation” in the Figure), provides better results than both the 1.3 and the 1.5 fixed incentive allocations over time. Using initial fixed incentives together with dynamic allocation definitively showed better results.

The results showed that using dynamic incentive allocation alone, without any initial fixed incentives (shown as “Dynamic 1.0 + Reputation” in the Figure), provides better results than both the 1.3 and the 1.5 fixed incentive allocations over time.

Both experiments in the scenario showed that dynamic incentive allocation is better than a fixed incentive allocation.

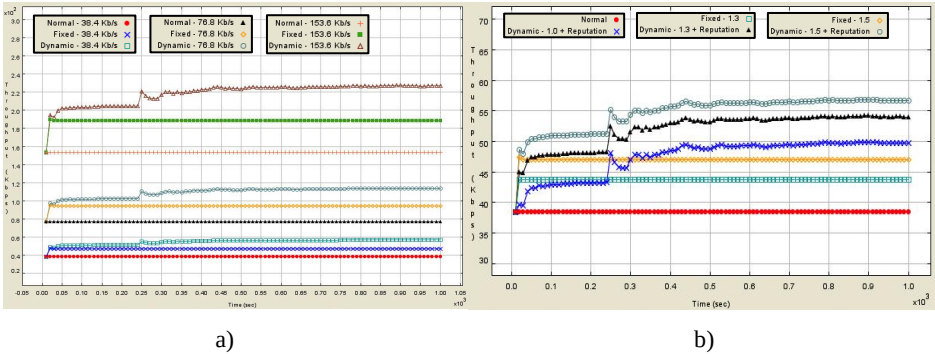


Fig. 1. a) throughput of relay-nodes with different low data rates; b) throughput of relay-nodes with different incentive weights

5 Conclusion

In this paper, we have proposed a dynamic reputation-based incentive mechanism that encourages users to participate in a cooperative relay service. Our experimental results showed the significant positive impact of using a dynamic reputation-based incentive mechanism in the simulation scenario examined. This suggests that it is very important to consider dynamic incentives for overcoming selfish behavior and increasing participation in PNs.

The proposed dynamic reputation-based incentive mechanism for cooperative relay can help to avoid situations in which relay nodes can renege after receiving their static or fixed reward of additional throughput. By this dynamic approach, we can enable fairer and more efficient allocation of scarce resources such as the radio spectrum.

Based on these findings, we argue that spectrum management decision makers should consider facilitating the adoption of such an approach, as this approach can provide more flexibility for MOs and can enable them to offer new value-added services and to create new business models.

References

1. Niemegeers, I.G., de Groot, S.M.H.: From personal area networks to personal networks: A user-oriented approach. *Wireless Personal Communications* 22(2), 175–186 (2002)
2. Wei, H.-Y., Gitlin, R.D.: Incentive mechanism design for selfish hybrid wireless relay networks. *Mobile Networks and Applications* 10(6), 929–937 (2005)
3. He, Q., Wu, D., Khosla, P.: A secure incentive architecture for ad-hoc networks. *Wireless Communications and Mobile Computing* 6(3), 333–346 (2006)

User-Centric Prediction for Battery Lifetime of Mobile Devices

Joon-Myung Kang, Chang-Keun Park, Sin-Seok Seo,
Mi-Jung Choi, and James Won-Ki Hong

Dept. of Computer Science and Engineering, POSTECH, Republic of Korea
{eliot,pck1982,sesise,mjchoi,jwkhong}@postech.ac.kr

Abstract. Today, mobile devices are being used for various applications such as making voice/video calls, browsing Internet and so on. The operating time and battery consumption spent in those activities affect the battery life of mobile devices. In this paper, we propose a method for predicting the battery lifetime of mobile devices based on usage patterns. We define the possible states of mobile devices based on their operating functions and develop a method of predicting battery lifetime based on average battery consumption and duration of each state.

Keywords: Battery Lifetime Prediction, Usage Pattern, Power Management.

1 Introduction

As mobile networks and services have been growing, mobile and battery-powered devices (e.g., a mobile handset or PDA) have emerged as important tools in our lives. However, their short battery life is one of factors that cause much inconvenience or reduce their usefulness [1]. Mechanisms to provide long and stable battery life are needed to provide mobile communication networks that are more stable. To guarantee long battery life is to minimize battery consumption by reducing needless battery usage.

The key to successfully minimizing battery consumption is accurately predicting the battery consumption of mobile devices' various operating states. Much work has focused on hardware or operating system (OS) level predictions [3], model-based battery life predictions [2], and battery consumption by a CPU when executing a particular instruction [4]; furthermore, battery lifetime has been usually predicted by static battery consumption rate which manufacturers of batteries and mobile devices have determined via experiments. It has also been predicted by the arithmetic means of the consumed power. However, previous methods have had limitations in predicting battery lifetime in terms of users' characteristics because they did not consider usage patterns. That is, every user has a different mix of voice call, data communication, and video call usage. If we assume that the usage pattern of each user is different and that it will continue on a similar basis in the future, we can predict battery lifetime on an individual basis.

In this paper, we propose a method for predicting the battery lifetime of mobile devices based on usage patterns. We define possible states of a mobile device based on its basic operating functions. We measure the battery consumption and the operating time spent in each state. We present our method using usage patterns generated by the probability of the operating time of each state. Because the proposed method presents the battery lifetime by way of an on-line analysis based on usage patterns rather than an off-line analysis, this is more useful to users than previous methods. It enables effective battery usage and detects abnormal battery usage by comparing operating times between the normal and abnormal states.

The organization of this paper is as follows. Section 2 describes our proposed method with problem definition and solution approach. finally, conclusions and future work are drawn in Section 3.

2 Proposed Battery Lifetime Prediction Method

In this section, we present our method of battery lifetime prediction. We first present our assumptions and problem definition. We then describe our proposed method and evaluation.

2.1 Assumptions

For our proposed method, we have made the following assumptions.

- the battery consumption of mobile devices is affected by voice calls, video calls, data communication, short message service, LCD, applications, music, and waiting for calls
- we can define possible states in terms of the operating functions of mobile devices
- the average battery consumption rate of each state is different
- the probability of duration of each state is different because different users have different usage patterns
- if a users' usage pattern will be repeated and consistent over the long term, we can use Zipf's law to predict the battery lifetime, based on usage pattern

2.2 Problem Definition

We also assume that a mobile device has n possible states. First of all, some symbol descriptions and formulations are given as follows:

$$\mathbf{B} = (B_1, \dots, B_n), \quad (1)$$

$$\mathbf{p} = (p_1, \dots, p_n), \quad (2)$$

$$\mathbf{R} = (R_1, \dots, R_n), \quad (3)$$

$$R_i = p_i \cdot B_i, \quad (4)$$

where B_i is the average battery consumption rate of the i th state, and p_i time consumption rate of the i th state, which is determined to satisfy $\sum_{i=1}^n p_i = 1$. Because each user has a different vector $\mathbf{p}$ according to the usage pattern, this vector is defined for the symbol that represents the usage pattern. R_i is the ratio of battery consumption of the i th state.

Formulation to predict battery lifetime based on usage pattern:

$$T = \frac{V}{\sum_{i=1}^n R_i} = \frac{V}{\sum_{i=1}^n p_i \cdot B_i}, \quad (5)$$

where T is the battery lifetime predicted by usage patterns, and V is the total battery quantity. Because each user has a different $\mathbf{p}$, every user has different lifetime for the remained battery.

2.3 Solution Approach

In this section, we describe a solution of the problem defined in Section 2.2. When a mobile device is turned on, a data-collecting process starts to gather data *vis-à-vis* the operating time and battery consumption of each state. This process records the battery and time consumptions of each state, based on timestamp. This time-series data is used to measure average battery consumption rate ($\mathbf{B}$) and usage patterns ($\mathbf{p}$). Finally, the prediction of battery lifetime, based on usage pattern, is provided based on these two vectors. First of all, we collect the time-series data for each state. To collect data for each state, we append time and battery level to a log file upon entering and exiting the state. We define a symbol for log data in the vector $\mathbf{D}_j = (D_j^1, D_j^2, D_j^3)$, where $\mathbf{D}_j$ is a vector of the j th log data, D_j^i is the time-series data of the j th log data, D^1 is a state of the mobile device, D^2 is a battery level, and D^3 is time consumption. Next, we look to calculate the average battery consumption rate and the usage pattern using time-series log data. As defined in Section 3.2, we assume that a mobile device has n possible states and the log file has m entries. We represent the total battery consumption in the vector $\mathbf{b} = (b_1, \dots, b_n)$. The value of this vector is the total battery consumption of the i th state via a summation of the second elements of each log entry which state is i . We also represent the time consumption in the vector $\mathbf{t} = (t_1, \dots, t_n)$. The value of this vector is the time consumption of the i th state via a summation of the third elements of each log entry which state is i . Thus,

$$b_i = \sum_{j=1}^m D_j^2, \quad (6)$$

$$t_i = \sum_{j=1}^m D_j^3, \quad (7)$$

where $D_j^1 = i$. The average battery consumption rate and usage pattern for each user using Equations (6) and (7) are defined as $B_i = \frac{b_i}{t_i}$, $p_i = \frac{t_i}{\sum_{i=1}^n t_i}$, where B_i is the average battery consumption rate of the i th state, and p_i time

consumption rate of the i th state, which is determined to satisfy $\sum_{i=1}^n p_i = 1$. This usage pattern represents not only the usage pattern of the past, but also the probability of spending time in each state in the future. We can predict the usage pattern using Holt-Winters Model [5,6] which predicts the future data based on time-series data. So, we can calculate the lifetime of remained remaining battery (T) by Equation (5) with $\mathbf{B}$ and $\mathbf{p}$.

3 Concluding Remarks

Predicting of the battery lifetime of mobile devices is important to successfully minimizing battery consumption at the application level. In this paper, we have proposed a prediction model based on usage patterns, such as the battery consumption rate spent in making voice calls, using data communication, or waiting for calls. We have also presented the usefulness of our proposed method as a case study. The contribution of our proposed method is to provide a stable mobile communication network through the user-centric prediction of mobile device use.

Currently, we are testing our proposed method on a mobile platform based on Intel PXA270 processor based on the ARMv4 instruction set architecture. We have already ported the embedded Linux, kernel version 2.6.15.7; because we can modify the device driver for battery and voice call on the embedded Linux, we can derive more precise values. We will present the experimental results that validate our proposed method on this mobile platform, in the extended version of this paper.

Acknowledgments. This research was supported by ECED at POSTECH under the BK21 program of Ministry of Knowledge Economy (MKE), Korea and under the ITRC support program supervised by the IITA of MKE (IITA-2008-C1090-0801-0045)

References

1. Krintz, C., Wen, Y., Wolski, R.: Application-level Prediction of Battery Dissipation. In: Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED 2004), Newport Beach, CA, USA, pp. 224–229 (August 2004)
2. Doyle, M., Fuller, T.F., Newman, J.: Modeling of Galvanostatic Charge and Discharge of the Lithium/Polymer/Insertion Cell. *J. Electrochem. Soc.* 141(1), 1–9 (1994)
3. Compaq, Intel, Microsoft, Phoenix, and Toshiba: Advanced Configuration and Power Interface Specification (2002)
4. Tiwari, V., Malik, S., Wolf, A.: Power Analysis of Embedded Software: A First Step towards Software Power Minimization. In: *IEEE Transactions on VLSI Systems*, pp. 437–445 (December 1994)
5. Holt, C.C.: Forecasting Seasonals and Trends by Exponentially Weighted Moving Averages. Carnegie Institute of Technology, Pittsburgh (1957)
6. Winters, P.R.: Forecasting Sales by Exponentially Weighted Moving Averages. *Management Science* 6, 324–342 (1960)

A Study on the Reliable and Flexible Implementation of Workforce Management Using Business Process Management

Kyu-Hwal Kim, Byeong-Yun Jang, Min-kyu Kwon, and Seung-Hak Suk

KT Network Technology Laboratory,
463-1, Jeonmin-Dong, Yuseong-gu, Daejeon, Korea
{netace, bychang, kwon2mk, suksh}@kt.com

Abstract. In recent years, global telecommunication service providers are struggling to survive through service convergence, integration, and bundling. In these circumstances the telcos hope to maintain their network stable but catch up with the brand-new business trends. Therefore they also want to diagnose the process related to the network operations management and improve it into the optimum. In this paper, the flexible and reliable architecture of workforce management, which is part of the network operations management, will be studied as follows: First, we will mention the business process of workforce management related to service assurance in telecom sector. Second, we design and model the workforce management's activities according to the Business Process Management (BPM). And the specific implementation is discussed. By adopting the BPM, the network service providers can track and monitor their current status of workload, workforce. They can modify their workforce management into more easily integrated and process-based architecture.

Keywords: Business Process Management, workforce management, service assurance.

1 Introduction

As we all know, the telecommunication market has almost reached its saturation point and the increase of subscribers has been slowing down. The telecommunication Service providers offer a bundle of services together, for example, telephone, Internet, and TV(so called triple play service, TPS). In the meanwhile they need to maintain stable network status with minimum operational cost for providing reliable services. It is inevitable that we need a new network operations and management paradigm to adapt easily to the change of business and operations environments [2].

The importance of workforce management, which is part of network management and operation systems, is also increasing along with the speedy business environment. Business process management (BPM) is a systematic approach to improving an organization's business processes. In this paper we describe the process flow related with the workforce management involving service assurance, before service(BS) checkup with regard to the network operations management. And we will do process modeling regarding its network operation activities by BPM tools.

2 Business Process Design of Workforce Management

KT's business process associated with workforce management for the purpose of network operations & management is composed of two main activities namely, service assurance and before service. Figure 1 shows the process flow about the service assurance and before service. In the case of before service, we start the yearly checkup planning. But In the service assurance we start by receiving the trouble ticket which is generated by NMS or customer claims. A manager(or automated scheduling module) goes over the status of field technicians by considering their ability, skills, turfs, locations, day off or not. And he arranges the appointment schedule and finds a suitable worker. Thereafter he will assign technician to work order and send the work order by mobile environment. Then the technician receives the workload and decides whether he will do or not. If there is more suitable person in the work order or he cannot do the job, he asks to re-assign the job. When jeopardies happen in executing step, a manager or a field technician would cancel, suspend and reschedule the field order. In the normal cases, a field worker executes and completes the job. In the reporting step, he reports the result of checkup or troubleshooting. Finally workforce management system closes the field order and analyzes data like average lead time, level of difficulty of workload, the skillfulness of individual technician and so on [5].

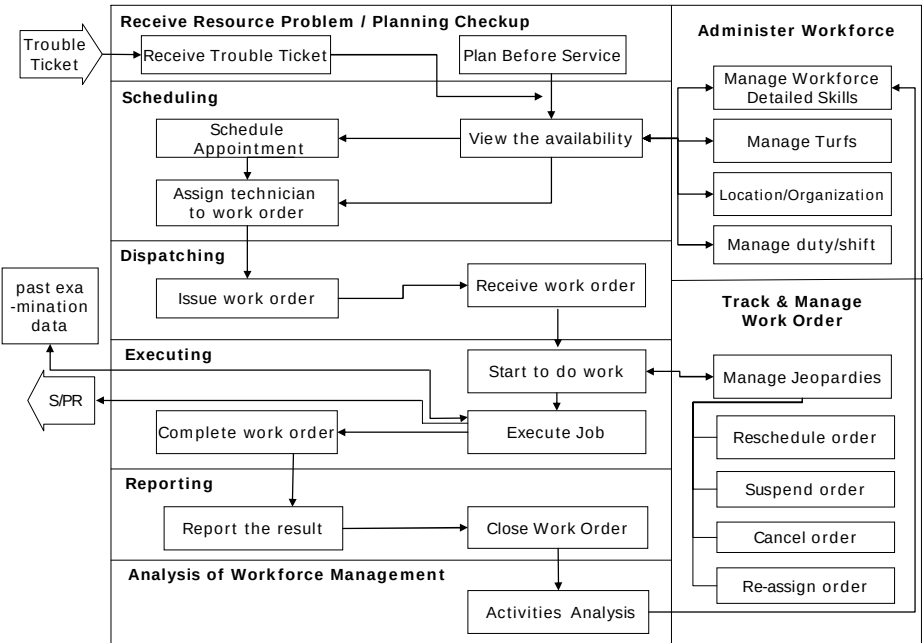


Fig. 1. The process flow of workforce management

Based on the process flow we implemented the process model in the planning step by the Websphere Modeler, which is IBM BPM suite, as Figure 2. The process model in the planning step shows that the process is separated into the trouble and BS checkup plan.

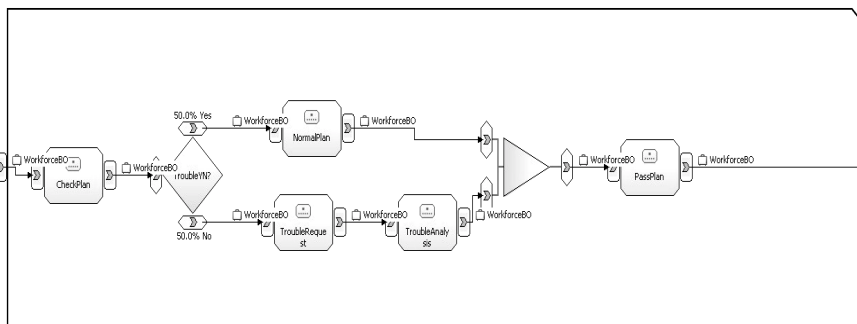


Fig. 2. The process model in the planning step with Websphere Modeler

We convert the process model into BPEL process model. Business Process Execution Language(BPEL) is a XML based language, which is able to orchestrate web services to a business process. Figure 3. shows the BPEL process model of the overall steps(but execution stage excluded).

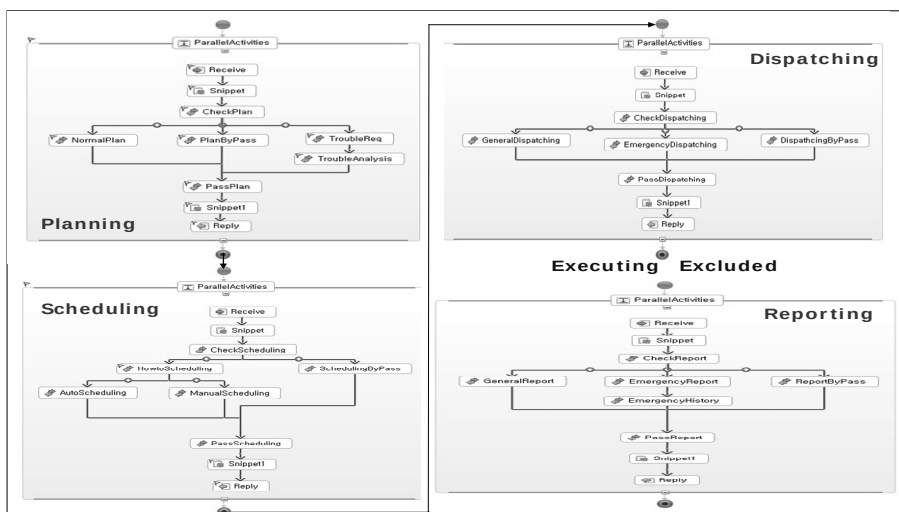


Fig. 3. The BPEL process model in the overall steps(without Executing stage)

Figure 4 shows the linkage mechanism between BPEL and Application Component in the Workforce Management System. We mediate the application service component with the BPEL process through the TaskConductor Service Module, which is web-service-based linkage component. The Task conductor is composed of TaskReceiverWSExport, TaskReceiver, TaskConductor, TaskLogger. The TaskReceiverWSExport receives the result from the web application as input interface, transfer it to TaskReceiver. The TaskReceiver invokes the TaskConductor component. The TaskConductor perform the BPM process, the TaskLogger logs the result of BPM process and The TaskConductor returns its result to the TaskReceiver. The TaskReceiver passes it to the Application Component which invoked it.

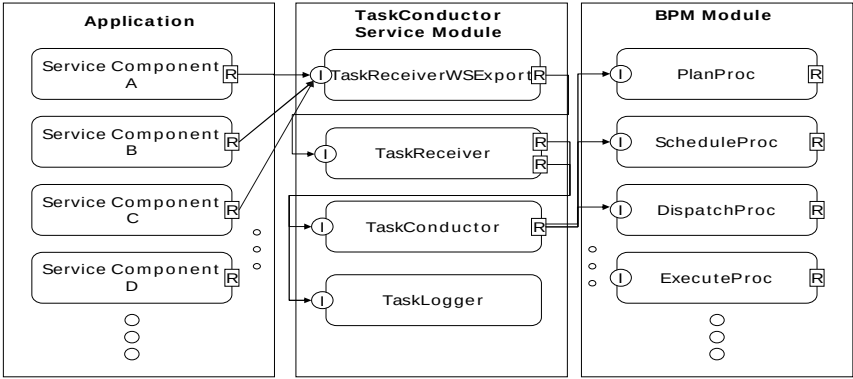


Fig. 4. The linkage mechanism between BPEL and Applications

The service components in application pass the parameters like the verb, business process Instance ID, BPM user ID, HumanTask Instance Name and receive the result of BPM process, error messages, timestamps etc. The BPM tool provides the time stamp of each process instance from planning stage to reporting. The workforce management system gets the time information , evaluates the overall processes and finds the bottleneck. Based on the data, we can adjust the current process model and regenerate the BPEL. We can implement the workforce management system flexibly and reliably.

3 Conclusions

In the competitive business environments, the business process is likely to change for the customer's satisfaction. Therefore the operations support systems(OSS) should be modified quickly and reliably. In this paper we studied the implementation case of workforce management system in KT by using IBM BPM suites.

References

- [1] Kwon, M.K., Suk, S.H., Jeon, D.Y., Chung, B.D.: A Study on the Architecture of the Field Operation Support System. In: KNOM Conference 2008 (April 2008)
- [2] Bang, Y.-H.: NOM 2.0: Innovation Network operations and Management for Business Agility. IEEE Communication Magazine (March 2008)
- [3] Reilly, J.P., Creaner, M.J.: NGOSS distilled. Cambridge University Press, Cambridge (2005)
- [4] Muehlen, M.z.: Workflow-based Process Controlling (June 2004), ISBN 3-8325-0388-9
- [5] Jose, R.: The workforce manage decomposition proposal. TMForum (December 2000)

Enterprise Management System with Web-Crawler

Myung Sil Choi, Yong Soo Park, and Kwang Seon Ahn

Department of Computer Engineering,
Kyungpook National University at Daegu, Korea
{silchoi, timpark75, gsahn}@knu.ac.kr

Abstract. Corporations offer their PR activities, voices of customers, and other diverse information over their Web sites. In this process of information provision, the Web administrator's negligence or customers' negligence occasionally exposes the private information over the Web sites. This paper proposes a system to detect the location of private information in such Web pages, thus defining vulnerable areas. For this purpose, we use a kind of a transformed crawler method.

1 Introduction

A corporate Web site is a means aiming to publicize the corporation and to communicate with customers. While a corporation endeavors to provide a greater quantity of information over its Web site, the Web administrator may inadvertently expose private information from the site. Currently, powerful Web robots (also called Wanderers or Crawlers) automatically download and search a vast number of web pages. However, the more powerful a search engine (such as Google, Yahoo) [1, 2] is, the more customers' private information is frequently exposed. Our proposed Enterprise Management System (EMS) aims to determine the amount of private information exposed in corporate Web sites thus detecting areas vulnerable to exposure.

In this study, first, we use a transformed Web-crawler-based search method to collect private information from the limited space of corporate Web sites. Second, we propose a method to resolve the problems of hidden hyperlinks [3] searched over Web sites, and thus to detect private information more closely.

2 EMS System Overview

The proposed EMS is a Web-crawler-based management system designed to detect private information exposed in corporate Web sites. Figure 1 shows the structure of EMS. EMS consists of two steps from the Web Crawler Engine (WCE) and a Risk Analysis Engine (RAE), respectively: The system collects web pages from WCE, and searches private information exposed from RAE. A WCE can access only the Web server of the joint enterprises. The Web server of the relevant enterprises must allow the Web Crawler engine to approach "robot.txt" in accordance with Robot Exclusion Standard.

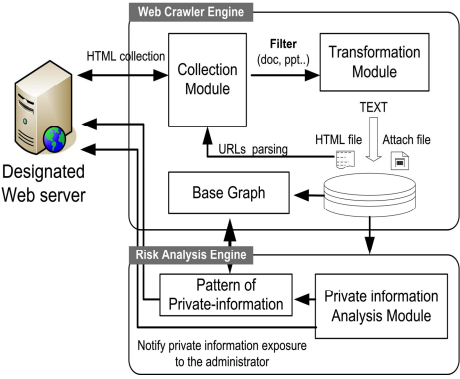


Fig. 1. Architecture of EMS

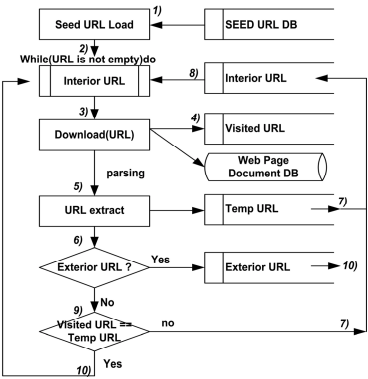


Fig. 2. Collection Process

2.1 Collection Module

The Web crawler uses the page link, gets URLs of Web pages from the seed URL, and download them again, thus repeatedly collecting Web pages. In Figure 2, the process of collecting Web pages belonging to the collection module is outlined as follows. 1) Read URLs from the “Seed URL DB” and initialize the Seed URL. 6) Compare URLs of Web documents and parsed URLs; if the domain name is different, then register the URL with the “Exterior URL DB. 7) If the domain name is identical, register the “Temp URL” with the “Interior URL DB.” 8) Sequentially select new “Interior URLs,” and repeat the process of 2) ~ 5). 9) When extracting and analyzing the collected documents, if the “Visited URL” is identical with the “Temp URL” or is “Exterior URL,” 10) then progress sequentially with the “Interior URL.” If URLs no longer exist in the “Interior URL,” then the process will end. This paper examines a system to collect Web pages from a Web site and analyze them, thus avoid considering Exterior URLs.

2.2 Transformation Module

In particular, the first portion of this system has a Transformation Module(TM). The TM transforms hidden hyperlinks or attached files into text files. A hidden hyperlink refers to a case where with no dynamic URL included in the Web page, the link page address cannot be accessed (e.g., Javascript:moveUrl('2','0','935260','9682099','0','157 67') of JavaScript). Also, attached files include flash and other multimedia files and binary files. Actually, private information can be included in attached binary files with great possibility. Our proposed system addresses the problem of hidden hyperlinks, and converts binary files into text and analyzes them. This process can extract a greater amount of exposed private information, and collect link information (URLs) contained in binary files. However, this process increases search time in proportion with the number of link information collected.

2.3 Private-Information Analysis Module

The Private information Analysis Module analyzes the private information pattern of downloaded documents and Web pages. These private information patterns are based on Java's regular expression. For instance, the pattern can be expressed as

Social-Security-Number=(\d{6})([^\s]+([1|2]\d{6}\D);

Cellular-telephone-number=(010[^\s]{0,3}\d{3,4}[^\s]{0,3}\d{4}); .

Logs (URLs) collected from WCE are expressed in terms of a base graph to manage EMS. They also conduct the risk of continually changing Web sites. The base graph expresses Web documents into directed graphs, and the Collection Module basically conducts crawling with the Depth First Search (DFS). RAE analyzes the exposure frequency of private information, and then begins crawling from the node with the greatest frequency exposure (URL or Web document) on the Base Graph.

3 Experiment and Implementation

In this Section, a simulation test is conducted to determine how much private information our proposed EMS can collect. Actually, our system has no frameworks to compare it with. Thus, we compare the previous DFS method and the DFS containing the transformation process (called TM-DFS).

3.1 Experiment Setup

In the test, we collect Web documents from the "http://www.ce.knu.ac.kr" site. The EMS System Web Crawler engine demonstrates the Java language. The collected attached files were designated as different file types (e.g., ppt, pdf, doc, hwp, zip, xls, etc.). The collection time was unlimited until the collection depth was over. The number of threads was found to reach five, at random, for the purpose of maximizing the period of the Web server Web access. The thread check time was found to reach 500 ms, and the thread timeout was found to reach 600,000 ms.

3.2 Experimental Evaluation

We compare the existing DFS method and the TM-DFS method. Table 1 shows those data that were collected from WCE with TM with page depth set at 1, 5, and 10. Figure 3 shows a comparison of the number of documents collected by DFS and TM-DFS with page depth set at 5. The number of link URLs collected by TM-DFS is 1.46 greater than that collected by DFS, because it addressed the problem of hidden hyperlinks and crawled link URLs in binary files. Also, TM-DFS exposed the number of information 1.18 times more than DFS did. The findings confirm that private information exposed in the Web site was sufficiently detected.

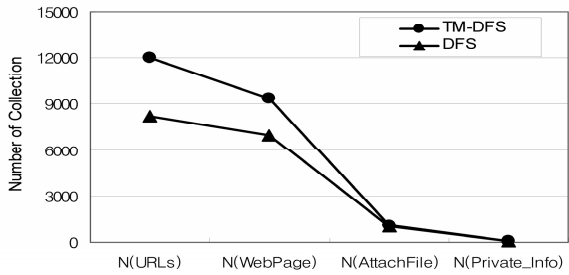


Fig. 3. Comparison of the number of documents collected by DFS and TM-DFS

Table 1. Valuation in accordance with Page depth

Page Depth	Number of Link URL	Number of Web page	Number of Attach file	Number of Private information
Depth-1	48	42	1,102	4
Depth-5	11,994	9,370	1,102	47
Depth-10	13,742	9,526	1,117	49

4 Conclusion

In this study, we implemented a system to extract private information exposed in corporate Web sites. For that purpose, we used a kind of transformed crawler method. Our proposed method addressed the problem of the inability to trace dynamic URLs such as JavaScript URLs, collected link information contained in binary files, and thus sufficiently detected exposed private information.

References

- 1. Brin, S., Page, L.: The Anatomy of a Large-Scale Hypertextual Web Search Engine. In: Proceedings of the 7th International World Wide Web Conference, Brisbane, Australia (1998)
- 2. Cho, J., Shivakumar, N., Garcia-Molina, H.: Finding Replicated Web Collections. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, Dallas Texas (2000)
- 3. Raghavan, S., Garcia-Molina, H.: Crawling the Hidden Web. In: Proceedings of the 27th International Conference on Very Large Databases, Rome, Italy 2001)

A Construction Process for Small-Scale Network Systems

Yuka Kato

School of Industrial Technology, Advanced Institute of Industrial Technology,
1-10-40 Higashi-Ohi, Shinagawa-Ku Tokyo 1400011, Japan

Abstract. This paper proposes an efficient construction process for small-scale network systems, including name services, mail services and so on. In the research field of software engineering, many studies have been made of software development processes, and high quality software systems have been efficiently developed by using them. The proposed scheme applies software development processes to network system construction, and defines the process which consists of "requirements", "design", "implement", "testing" and "deployment" processes. This paper also shows an application result of the proposed scheme to a network course at a graduate school.

Keywords: Network systems, network design, software development processes, small-scale networks, process management.

1 Introduction

Various kinds of knowledge are needed to construct network systems including name services, mail services, and so on. Recently, secure network systems are especially required, and a lot of know-how exists for such system construction tasks. However, there are no systematic processes, and people's experiences are required for the tasks. This is because network systems have various network scales and system environment, and it is difficult to develop a system of the construction processes. On the other hand, in the software development area, many studies have been made of software engineering[1][2][3], and software development based on the given processes is usually done. This paper applies these software development processes to network system construction, and proposes the construction process for network systems.

2 Scope of the Proposed Scheme

The target in this paper is small-scale network systems, and the following is a specific example of the target network systems.

- To construct and maintain a network system for a small-scale organization in which the number of members is about 10.
- To assign one global IP address by an ISP and to use PC servers.
- To construct four types of systems. These are an access router, an internal router, external servers (DNS etc.) and internal servers (DHCP etc.).

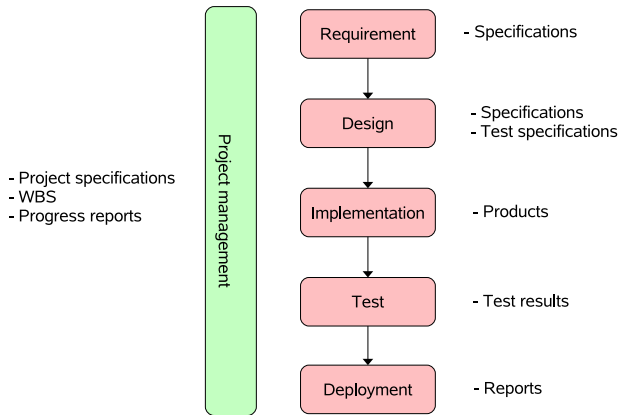


Fig. 1. Construction process for network systems

3 Construction Process for Network Systems

In this section, the construction process, which consists of requirements, design, implement, testing and deployment, is defined based on the software development processes (See Fig. 1). Some number of iterations are adopted if necessary.

3.1 Requirements

In this process, system requirements are investigated and the requirement specifications are drawn up. There are no templates of the documents in it.

3.2 Design

This process corresponds to the software design task. The purpose of the process is to draw up design specifications, and it consists of the following three parts.

Network system design: Network systems are designed to fulfill the requirements. At that time, designers use document templates and fill in the forms to design the systems. The following are concrete specifications to be made.

- Plans of logical and physical network architecture, and firewall policies
- Plans of configurations on the access and internal router
- Plan of configurations on the external and internal servers

Security design: Designers determine the security policies, and pick up necessary tasks for constructing a secure system. The designing results are used for a part of the system design. The following are some examples.

- All of ports which do not provide any services are closed.
- The security managers are designated, and they monitor system logs.

Testing design: Designers make test cases of the servers and routers according to the design specifications, and draw up test specifications. At that time, they use document templates and fill in the forms for the testing design.

3.3 Implementation

There are three products (the operational plans, the implementation results and the operational logs) in this process. Because it is difficult for non-experts to estimate the volume of tasks, it is better to implement simple functions briefly at the early stage, and then to modify the whole schedule.

3.4 Testing and Deployment

The person in charge of testing conducts tests for the items described in the specifications. After that, the project documents are completed.

3.5 Project Management

In order to manage all construction processes, project management is needed. In this task, before requirement definitions, the project definitions and the specifications are made. The master schedule has to be defined in the first stage, and progressive elaboration should be done according to the progress.

4 Evaluation

This section shows the construction results by four teams, which have 4 or 5 students of a graduate school. The project requirements were the construction of the system described in Section 2 (the period was 180 min. x 8 weeks).

4.1 Evaluation Metrics

The following evaluation metrics are defined in this paper; (1) Does the procedure cover all necessary tasks? (2) Is the system secure? (3) Quality of products and documents, (4) A gap between plans and results.

4.2 Construction Procedure of Each Team

In the course, the instructor specified the requirements, and the student teams designed the project plans. The following are the actual procedures.

Team A: They drew up the master schedule with four iteration cycles, and two cycles were actually conducted. Concrete implemented services were the access router and the external servers (SSH, DNS, SMTP, WWW).

Team B: They adopted iteration processes, but they did not draw up the master schedule and added a new iteration cycle as progressing the work. Concrete implemented services were the external server (SSH just from a LAN) and the internal server (DHCP).

Team C: They did not use iteration processes and adopted a waterfall model. Concrete implemented services were the access and internal routers and the external (SSH, DNS, WWW) and internal (DNS, SSH) servers.

Team D: They did not use iteration processes and adopted a waterfall model. Concrete implemented services were the access and internal routers and the external (SSH, DNS, WWW) and internal (DNS, DHCP, SSH) servers.

4.3 Evaluation Results

Table 1 shows evaluation results. Three instructors were in charge of the course, and evaluation values of metric (2) and (3) are the average of their evaluation results according to the five-grade system. The value of (1) and (4) are given by the author based on the procedure also according to the five-grade system.

Table 1. Evaluation results

Team	(1)	(2)	(3)	(4)	process
A	5	4.7	4.0	5	iteration
B	4	3.0	2.8	2	iteration
C	4	4.0	3.8	3	water fall
D	5	4.0	4.2	3	water fall

4.4 Discussion

From the evaluation values of (1) and (2), we found that the definition of construction processes makes it possible for all teams to design reliable network systems. However, from the evaluation values of (3) and (4), we also found that there are some differences of quality of products and the gap between plans and results among the teams. Here, in the teams where iteration processes go well (that is team A), the quality of the products is high. It is the effect of the iteration model. Team B also adopted iteration processes, but they did not draw up the master schedule and the processes did not go well. In team C and team D, member’s experiences affected the quality of the products. Since it is difficult to estimate work volumes under various system environments, the effectiveness of iteration processes would be large even for small-scale network systems.

5 Conclusion

This paper applied the software development processes to network system construction tasks, and proposed a construction process for small-scale network systems. The author also conducted implementation experiments for the proposed scheme. As a result, we found that the process definition made it possible to construct reliable systems efficiently within a limited period of time.

References

1. Pressman, R.: Software Engineering: A practitioner’s Approach. McGraw-Hill, New York (2004)
2. Pfleeger, S.R.: Software Engineering. Prentice-Hall, Englewood Cliffs (2005)
3. Booch, G., Maksimchuk, R.A., Engle, M.W., Young, B.J., Conallen, J.: Object-Oriented Analysis and Design With Applications. Addison-Wesley, Reading (2007)

Design, Implementation and Evaluation of a Network Management System for a High Quality IP-Based Video Transmission Service

Shuntaro Kashihara, Kenichi Ogaki, and Tomohiro Otani

KDDI R&D Laboratories Inc.

2-1-15 Ohara, Fujimino City, Saitama Prefecture, 351-8502, Japan
sh-kashihara@kddilabs.jp

Abstract. To meet the growing demand for IP-based video transmission services, we investigated the architecture of a Network Management System (NMS) from the viewpoint of actual operations and management. To verify the architecture investigated, we designed and implemented a prototype of the NMS, which manages both the video transmission service and the underlying network infrastructure based on Multi-Protocol Label Switching (MPLS) Diff-Serv-TE technology. The developed NMS successfully enabled operators not only to associate video service sessions with Label Switched Paths (LSP) and network equipment, but also to immediately assess the influenced services and LSPs by pinpointing network failures.

1 Introduction

Thanks to recent progress in IP networking technologies and an IP-based video encoder and decoder (codec), video signal transmission over the IP network has been already commercialized in the form of IP-TV services to consumers and there is growing demand for economical, high quality real-time video and stored material video transmission via the use of IP networking technology.

However, such video transmission has strict quality requirements 1, meaning private leased lines based on ATM or SONET/SDH have been mainly utilized to date, increasing the cost and making it difficult to upgrade the bandwidth of services. To achieve low-cost and flexible video transmission services, Multi-Protocol Label Switching (MPLS) DiffServ-TE technology is one of solutions, which is able to guarantee the bandwidth over the IP network. Here, each video streaming service is provided and controlled by a single Label Switched Path (LSP). By changing the bandwidth assignment of LSPs, the services can be featured and network resources managed between the sending and receiving stations 2.

In order to promote such video transmission services over IP networks, the establishment of management and operation functions is the key within a realistic environment. In this paper, we investigate the architecture of a Network Management System (NMS) from the viewpoint of actual operation and management.

2 Design and Implementation of the NMS

We assume three types of services; namely real time video with the highest priority, stored video with medium priority and best effort data traffic with the lowest priority. Those services are assigned to MPLS LSPs in a one-to-one manner and separately forwarded based on LSP's priority. Real time video services are controlled so as to be assured with the specified bandwidth determined by the IP codec's compression ratio and the definition of the video. Even if the total ingress traffic amount surpasses the available egress bandwidth at the node, real time video services are designed to be forwarded and maintained with consistent performance, while best effort traffic services with the lowest priority are dropped. The video stream carried by the assured LSP is managed as the video session.

We designed and implemented a prototype of the NMS based on the three types of functionalities which we defined to manage, operate and control these IP-based video transmission services and their underlying network as shown in Fig. 1. And the NMS visualizes the network topology and network defects based on the database and helps operators conduct the network operation procedures more easily. In this section, we describe these functionalities of the NMS in detail.

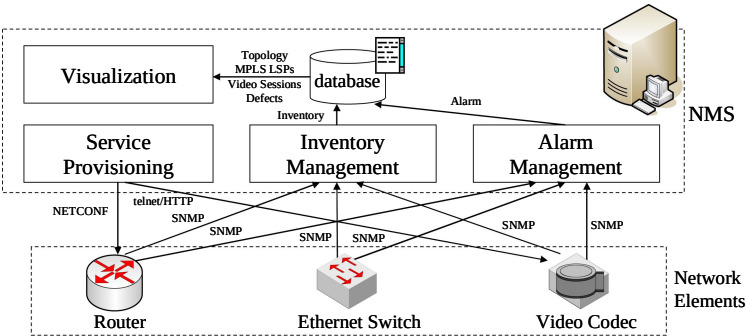


Fig. 1. Architecture of the NMS

Inventory Management Functions: When the NMS is initially installed or booted up for a reason, it is designed to discover network elements and search for the network topology. The NMS pings or telnets network elements, registers their information, and collects the necessary MIB information. The NMS achieves connectivity among IP/MPLS routers based on the Traffic Engineering (TE) link information collected through NETCONF over SSH 3, which is the eXtensible Markup Language (XML) based remote configuration method. Figure 2 shows the network configuration of our network testbed for the development of the NMS. The network testbed comprises four routers, four Ethernet switches and two IP-based video codecs. Figure 3 shows the topology map in the NMS. Only TE links are always indicated in the map, as gray lines, while LSPs and video sessions appear as colored lines only on demand in Fig. 3. Figure 3 (a) indicates the example displaying one of the created LSPs. Solid and dashed lines connecting routers express working and protecting paths, respectively. Likewise, a video session can be displayed from the ingress codec to the egress

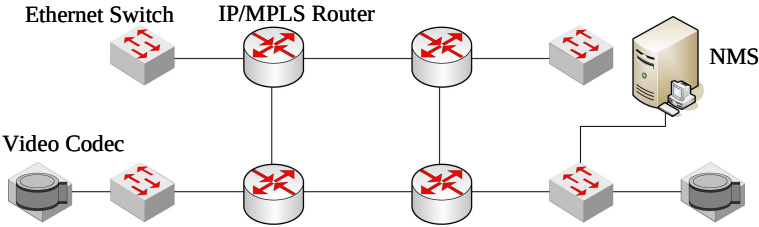


Fig. 2. Network Configuration of our Network Testbed

codec as shown in Fig. 3 (b). The relationship between the video session and the LSP are resolved by the multilayer management method 4. The next hop IP address of the service corresponds to the interface address of the adjacent IP/MPLS router through the Ethernet switch, and the MPLS LSP information related to the interface address is obtained from the route table information in the IP/MPLS router. The relationship between the Ethernet switch and the video session is resolved by comparing the IP addresses of the VLANs for management with the IP address space of the IP-based video codecs, and the NMS can appropriately display the route of the video session.

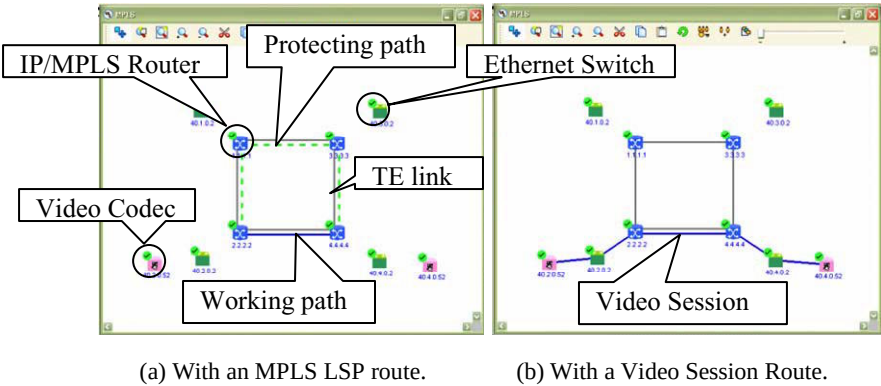


Fig. 3. Examples of Topology Map in our NMS

Service Provisioning Functions: we show the service provisioning function of our NMS, such as the creation and deletion of LSPs and video sessions and modification of the LSP configuration. The operator is asked to enter parameters such as the source and destination router's IP addresses, service priority, required bandwidth and route information. Once those parameters are filled out, the actual LSP is provisioned onto IP/MPLS routers. The graphical user interface (GUI) helps operators avoid mis-configurations or human errors in operation, otherwise, the operator would need to configure tens of lines through the command line interface (CLI). Furthermore, the NMS is designed to automatically examine whether the LSP can be established appropriately. To configure the router from the NMS, NETCONF over SSH3 is utilized. The NMS also supports multiple vendors' IP-based video codecs. When a video

session is provisioned between codecs, the operator is urged to enter the service bandwidth, service source and the destination IDs and IP addresses, and the LSP name accommodating the service. Subsequently, the NMS starts to access IP-based video codecs by telnet or HTTP depending on the codec vendors, and configure it according to the entered information. The NMS provides certain LSP management functions. The operator can switch over the LSP from the working path to the protecting path and change the bandwidth assignment of the LSP. The designed NMS can eliminate the direct CLI configuration and simplify the operation procedure with NETCONF over SSH and this kind of simple and automated operational procedure enables the operators to manage the large scale of the network without resulting in human errors.

Alarm Management Function: we exploit the alarm management functions of the NMS. If a network failure occurs, network elements are configured to send SNMP traps to the NMS. The NMS shows the list of alarms according to the traps received and displays a failure point on the map. Since the NMS manages the relations among network elements, LSPs and video sessions, the failure information is correlated among those objects. The affected link was highlighted and routers connected with the affected link changed the alarm status indicator.

Furthermore, if one of the ports in an Ethernet switch is broken, the router connected to the switch may not detect the failure. Since the NMS manages the topology of the network, LSPs and video sessions, it is designed to display the alarm sign on the LSP and video session related to the port of the Ethernet switch. These functions help operators quickly discover the service impacted and seek out the root cause, leading to prompt dispatch of individuals to resolve the failure.

3 Evaluation in the Field Trial

To evaluate the functions of our NMS, we conducted a field trial, constructing a video transmission network between Tokyo and Osaka in Japan. We deployed MPLS routers, Ethernet Switches, IP-based video codecs and the developed NMS. The NMS was allocated in Tokyo.

Next, we conducted an evaluation of the NMS functions. After we installed and booted up the NMS, the NMS started to detect the network elements via its inventory management functions and successfully registered all the information relating to them to its database. To confirm the functionality of the provisioning, the operator in Tokyo remotely provisioned LSPs and video sessions from Osaka to Tokyo. The real time video service was redundantly provided by multiple LSPs, routers and video codecs unlike in the cases of stored video and best effort data traffic and a stable transmission performance was obtained, even within the actual operational environment. When an intentional link failure occurred, the video signal was successfully restored and the NMS detected the network defect and successfully displayed it on the NMS map.

Through these experiments, we could confirm that the functions of controlling LSPs, managing and operating network elements and detecting network defects were appropriately operated as designed.

4 Conclusion

We investigated the NMS architecture from the viewpoint of actual operation and management. To verify the investigated architecture, we designed and implemented a prototype of the NMS, which manages both the video transmission service and the underlying network infrastructure, based on MPLS DiffServ-TE technology.

References

1. Naito, S.: Optimal JPEG2000 Encoder Mechanism for Low Delay and Efficient Distribution of HDTV Programs, DISPS-P1.9. In: ICASSP (2004)
2. Ogaki, K.: KDDI's vision on future media transport network architecture based on MPLS. In: MPLS 2007 International Conference (2007)
3. Wasserman, M.: Using the NETCONF Configuration Protocol over Secure SHell (SSH), IETF RFC 4742 (2006)
4. Miyazawa, M.: Multi-layer Network Management System integrated with a Network Planning Tool for IP/Optical integrated network, pp. 1–9. OFC/NFOEC (2007)

Availability in Peer to Peer Management Networks

Ouldooz Baghban Karimi¹, Saleh Yousefi², Mahmood Fathy¹,
and Mojtaba Mazoochi³

¹ CE Department, Iran University of Science and Technology

² CE Department, Urmia University, Iran

³ Iran Telecommunication Research Center

ouldoozbk@comp.iust.ac.ir, s.yousefi @ urmia.ac.ir,
mahfathy@iust.ac.ir, mazoochi@itrc.ac.ir

Abstract. Being one of the important factors of evaluation in management networks, availability of a management system is a point of our focus. However this issue is evaded in the researches conducted on P2P management systems. In this paper, we investigate critical object data availability in peer to peer management networks from an analytical point of view. We show that time-based availability calculations gives more insights on availability when one is to design a management network, but while the system is online and working, presence-based availability would help to estimate the status of the management network more sensibly.

Keywords: P2P, Peer to Peer networks, availability, network management, NMS, service availability.

1 Introduction

Availability for a system is normally defined as the property of being reachable and useable upon demand by an authorized entity. It is numerated as the portion time that a system works correctly to the total time it is in use.

$$A = \frac{MTTF}{MTTF + MTTR} \quad (1)$$

There are several availability measurements for peer to peer networks: peer availability, average system availability, data object availability and service availability. Time-based availability [3, 5] is the availability calculated by a ratio of working time of an element to its total observation time. Presence-based availability [1] relies on the same aspect, except that it is based on the observation done by another element having a burden to put on the current element. So, if the element is down when it does not have to do a task, the down time will not affect its total availability in the system because it was not observed during this time. A comparison of time-based availability and presence-based availability has previously been done in [1]. In this paper we examine critical data object availability in P2P management networks from time-based as well as presence-based viewpoints.

2 Availability in Peer to Peer Management Networks

In this section, we first define a framework for the management network we have used. Then we formulate critical data availability in our management framework in the following section in two main classes, time-based and presence-based.

2.1 Management Framework

In [7, 8] there are some proposed architectures for P2P network management systems. Assuming our management network works on a structured peer to peer overlay, we also handle some kind of grouping in this p2p network as ruled by the following rules: TLMs are Top level managers who can have interface with the user and MLMs. MLMs are Mid-Level managers who can have interface with TLMs and LLMs. LLMs are Low Level managers who can acquire information form the managed objects and send them to MLMs

Our main goal in a management system is to gain information form the system, decide upon the information collected from the system and then to apply the changes to the system based on the decisions made upon the information. Information could only be collected from the managed objects by the LLMs. LLMs maintain a filtering on the information acquired and send them to the MLMs. MLMs could send them to TLMs if the user has requested them or if the decisions is to be made in the TLMs. MLMs can also work on a management by delegation method, i.e. they can be delegated by the TLMs to do some management tasks.

2.2 Critical Data Availability

Availability of a special block of data (Data Object) in peer-to-peer systems using m out of n erasure codes in a time-based calculation [2] is formulated as:

$$A_{Object} = P_o = \sum_{i=0}^{n-m} \frac{\binom{M}{i} \binom{N-M}{n-i}}{\binom{N}{n}} \quad (2)$$

P_o is the probability that an object is available, n is total number of fragments; m is number of fragments needed for reconstruction; N is the total number of machines in the P2P system; M is number of currently unavailable machines.

We consider that in our management network, each critical block of data is erasure coded in each level of our management system. In this system, most of the decision makings are done in the TLM layer. So when we need a critical data to make a decision, we have to obtain it from a managed object which is done in the LLM, then pass it to the MLM and then give it to the TLM. We use erasure coding at each level of managers. Availability of a critical block of data using time-based calculations would be a multiplication of availability at each level:

$$A_{CD} = \prod_{g \in \{LLM, MLM, TLM\}} \sum_{i=0}^{n-m} \frac{\binom{M_g}{i} \binom{N_g - M_g}{n-i}}{\binom{N_g}{n}} \times \prod_{l \in \{l_1, l_2\}} A_{Link}(l) \quad (3)$$

N_g is the total number of machines in each level; M_g is number of currently unavailable machines in each level and A_{link} is the Availability of the link between each group and its descendant.

Presence-based availability of a critical data in NMSs is quite challenging. Since we need all the information to be available to the user from the users' perspective and the user would only acquire information from the TLMs, we will need to consider the presence-based availability of the system and other parameters from the TLMs point of view. And if we use the m-of-n availability, equal to the erasure coding parameters we use for our data objects we would obtain the A_{CD} which would be:

$$A_{CD} = A_m^n = \prod_{g \in \{LLM, MLM, TLM\}} \frac{\int_{t=0}^T O_m(t)(N_g - M_g(t) - n)dt}{\int_{t=0}^T (N_g - M_g(t) - O_m(t))dt} \quad (4)$$

$O_m(t)$ is the online function of the group of peers with m members' availability needed to carry out a task. It returns 1 when m members in a group of peers are on and 0 otherwise. Other parameters are as defined above.

3 Numerical Results and Conclusion Remarks

We conducted a handful of numerical analysis using MatLab. With different numbers of m and n between 0 and 16 we found out the relevancy between the erasure coding parameter and availability for both time-based and presence-based calculations. We assumed different number of peers in the network with 30% to 70% of total nodes unavailable.

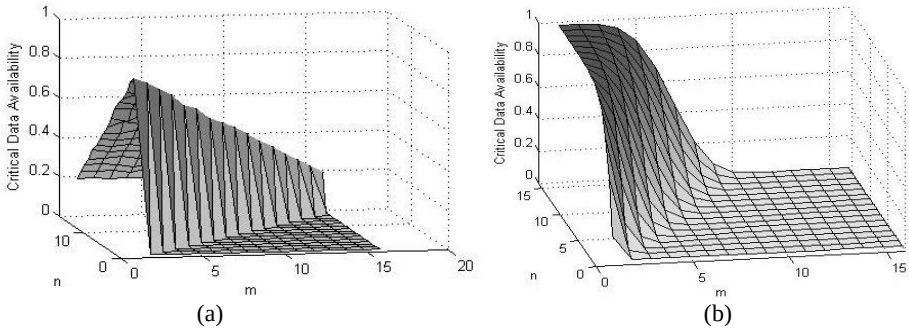


Fig. 1. (a) Time-based Availability for Critical data, with n and m varying between 0 and 16, (b) Presence-based Availability for Critical data, with n and m varying between 0 and 16

Time-based curve analysis shows us that the best availability could be gained when n is bigger than 10 and m is less than 8. We would gain a good availability of the system which in minimal value is equal to 99.01% when 40% of the total machines in the system are down.

On the other hand, on the same configuration, we could just gain 89.27% availability in the presence-based availability calculations, when n was 4 and m was 1 which means full redundancy. The reason for this result is that we considered that n in the m/n erasure coding is equal to the n in k -of- n presence based availability would gain us the critical data which is erasure coded on n machines. So m out of each group of n peers should be available to consider the critical data on that group considered to be available. $n=4$ and $m=1$ increases the chance of availability by redundancy.

The difference in the two cases is that the presence-based availability obliges all m machines out of n in the group because we know our data resides on that special group of machines to be online to get the availability required but time-based gives us no clue whether available machines carry the information we want or not. We conclude that time-based availability calculations would be more helpful when one is to design a management network because on that moment the designer has no information about the peers carrying the critical data, but while the system is online and working, presence-based availability would help to estimate the status of the network accurately and make management.

References

1. Dunn, R.J., Zahorjan, J., Gribble, S.D., Levy, H.M.: Presence-Based Availability and P2P Systems. In: Fifth IEEE International Conference on Peer-to-Peer Computing, pp. 209–216. IEEE Press, Los Alamitos (2005)
2. Weatherspoon, H., Kubiatowicz, J.D.: Erasure Coding vs. Replication: A Quantitative Comparison. In: Druschel, P., Kaashoek, M.F., Rowstron, A. (eds.) IPTPS 2002. LNCS, vol. 2429, pp. 328–338. Springer, Heidelberg (2002)
3. Prabhu, S., Venkat, R.: High Availability for Network Management Applications. In: Second International Conference on Availability, Reliability and Security, pp. 493–498. IEEE press, Los Alamitos (2007)
4. Da Costa, G., Marchand, C., Richard, O., Vincent, J.-M.: Resource Availability for P2P Systems. In: Proceedings of the 20th International Conference on Advanced Information Networking and Applications. IEEE, Los Alamitos (2006)
5. Ramanathan, M.K.: Increasing Object Availability in P2P Systems. In: Proceedings of the 18th International Parallel and Distributed Processing Symposium (IPDPS 2004), pp. 201–208. IEEE, Los Alamitos (2004)
6. Panisson, A., da Rosa, D.M., Melchioris, C., Granville, L.Z., Almeida, M.J.B., Tarouco, L.M.R.: Designing the Architecture of P2P-Based Network Management Systems. In: Proceedings of the 11th IEEE Symposium on Computers and Communications, pp. 69–75. IEEE, Los Alamitos (2006)
7. Granville, L.Z., da Rosa, D.M., Panisson, A., Melchioris, C., Almeida, M.J.B., Tarouco, L.M.R.: Managing Computer Networks Using Peer-to-Peer Technologies. IEEE Communications Magazine 43(10), 62–68 (2005)

Field Operations Management and Unmanned Operations Management toward NOM 2.0

Byeong-Yun Chang, Daniel Wonkyu Hong, Kyu-Hwal Kim, and Byung-Deok Chung

KT Network Technology Lab
463-1, Jeonmin-dong, Yuseong-gu,
Daejeon, South Korea
{bychang, wkhong, netace, bdchung}@kt.co.kr

Abstract. In this paper we study changes for telecommunication network operations and management environments. In addition, we review the concept of NOM 2.0 to cope with these environments changes. Finally, we introduce field operations management and unmanned operations management architectures for implementations of NOM 2.0 in KT. This paper is the first step of study for the implementation of NOM 2.0. We expect that several other researches and developments toward NOM 2.0 start referring to this research.

Keywords: NOM 2.0, telecommunication operations management, unmanned operations management, process management.

1 Introduction

The objectives of traditional network operations management monitor and control network infrastructures with minimum cost to provide stable services through maintaining stable network status [1, 2 and 3]. However, nowadays, the existing “stability-oriented” network operations management infrastructure and processes should be migrated into “service-oriented” and “business-driven” network operations management within the context of today’s multimedia and converged business environments. Toward this, the paper [4], NOM 2.0: Innovative network operations and management for business agility, introduces telcos’ new network operations and management paradigm in the converged world. Based on the concept of NOM 2.0, this paper proposes field operations management architecture and unmanned operations management architecture to implement NOM 2.0.

The organization of the remainder of this paper is as follows. In the next section we analyze the changes of telecommunication operations management environment such as business trends and decrement of operation personnel. We also briefly deal with NOM 2.0. In the section 3, we analyze the telecommunication operations management processes of KT. In addition, we make suggestions for improving the network operations management processes and build the field operations management system architecture. In the section 4, we analyze fault repair cases in KT and build the unmanned telecommunication operations management architecture. In the final section we summarize this research and introduce future research topics.

2 Background Study

In this section we analyze telecommunication business and market trends, and network operations and management trends. We also deal with the concept and the evolution direction of NOM 2.0.

2.1 Environment Analysis

As we all know, the current telecommunication market has almost reached its saturation point, and the increase of new subscribers has been slowing down. Consequently service providers need to find another profit creating market. As the result of this effort they offer triple play service (TPS) which is a bundle of services (telephone, internet and TV). A quadruple play service (QPS) is the more advanced convergence service by adding a mobile service. For service providers, QPS is the basis for a new business model. On the other hand, for customers, it presents a good opportunity to get a variety of multimedia services at lower prices.

2.2 NOM 2.0

NOM 2.0 builds automation and intelligence of network management to operate network with the minimum cost to cope with the change of operations environments. It also provides business agility to cope with the changes of technology and business environments. In addition, it guarantees the quality about end-to-end session in the next generation convergence environment. Moreover, it observes various kinds of world standards of network management perfectly, but it also aims to provide the tools to achieve the specific vision of service providers which are not defined in the world standards.

3 Field Operations Management

In this section we build the field operations management architecture of KT as an example of implementation of NOM 2.0. This is still an on-going project in KT. After complete construction of field operations management systems, KT could manage operation personnel and network resources most efficiently with minimizing the cost of coping with the changes to the operations environment.

To analyze network operations management processes in KT, we first need to classify the works that create values for customers. We did this with five levels.

For detail explanation of these processes, you can refer to the paper [5]. As an example of analysis of network operations management processes, the trouble management SOP in the power management center of KT has three time consuming processes that are provisioning and delivering resources, and report writing. The main reason of these bottlenecks is lack of IT support. Likewise after analyzing all network operations management processes of KT we evaluate the processes and make suggestions for the improvement.

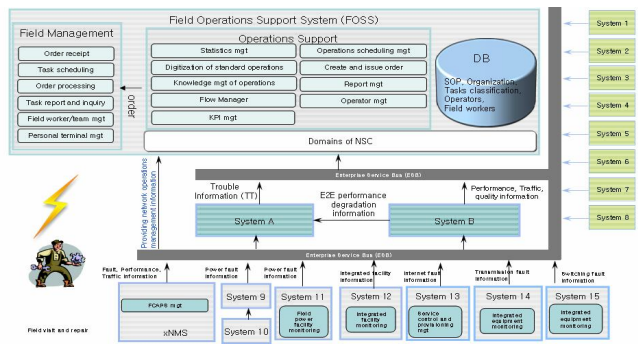


Fig. 1. Field operations support systems architecture

Based on the suggestions, discussions with the experts and background study in section 2 we plan to develop the field operations support system (FOSS). Figure 1 represents the system architecture of it.

4 Unmanned Operations Management

In this section we analyze fault repair cases in KT and categorize them. To reduce OPEX caused by on-site dispatch for fault repair, we introduce unmanned operations management architecture. This architecture could be useful to cope with the change of operations personnel such as continuous decrement of manpower in operations environment and needs of multi-skills.

The figure 2 in the next page represents the classification of fault and repair cases. In this figure, we analyze about 2,000 fault cases. Hardware and software errors occupied about 50% among all occurred faults.

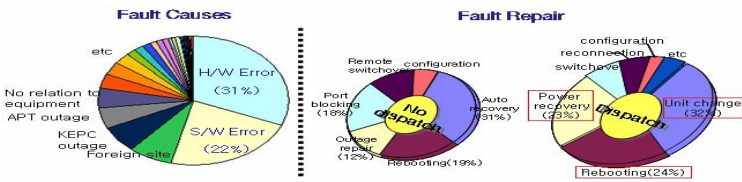


Fig. 2. Fault cause and repair case classification

To resolve fault problems, we can classify two cases. One is the dispatch case. The other is non-dispatch case. For the non-dispatch we can solve the problems without extra cost by dispatch repair. Thus if fault repairs by dispatch can be solved by non-dispatch, we could reduce OPEX. In the dispatch case, unit change and rebooting occupies most cases in fault repairs. Therefore, to change units and reboot equipment by non-dispatch we propose an unmanned operations management architecture in figure 3. This architecture is not a shot time vision but a long time vision about 15 years later.

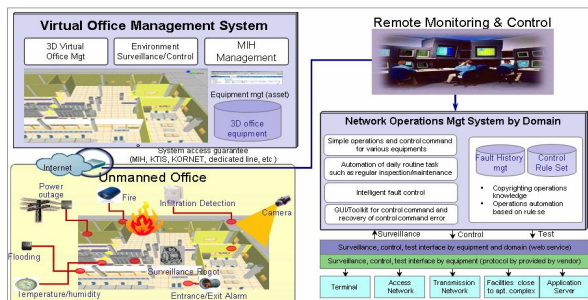


Fig. 3. Next generation unmanned operations management architecture

5 Conclusions and Future Research

In this paper we study telecommunication business and market trends, and network operations and management trends. We also review the concept of NOM 2.0. As implementations of NOM 2.0 we propose network operations management architecture and unmanned operations architecture.

For further research, straightforwardly, it is interesting to develop the system or the prototypes of the architectures in figure 1 and 3. It is also interesting topic of evaluating the introduction of the architectures economically. Network technology laboratory in KT is developing FOSS and realizing the unmanned operations management architecture.

References

1. Chang, B.-Y., Kim, H., Ko, S., Hong, D.: A study on service problem management and resource trouble management on a telecommunication network. LNCS, pp. 563–566. Springer, Heidelberg (2007)
2. Chang, B.-Y., Kim, H., Ko, S., Hong, D.: Next Generation Network Problem Management. KNOM Review 10(1), 32–42 (2007)
3. Reilly, J.P., Creaner, M.J.: NGOSS distilled. Cambridge University Press, Cambridge (2005)
4. Bang, Y.-H.: NOM 2.0: Innovative Network Operations and Management for Business Agility. IEEE Communications Magazine (March 2008)
5. Chang, B.-Y., Hong, D., Chung, B.-D.: Analysis of Network Operations Management Process in KT Technical Paper (2008)

The IMS/SDP Structure and Implementation of Presence Service

Jae-Hyoung Cho and Jae-Oh Lee

Information Telecommunication Lab,
Dept. of Electrical and Electronics Engineering,
Korea University of Technology and Education, Korea
{tlsdl2, jolee}@kut.ac.kr

Abstract. Nowadays in communication and network business, structure and system are independently constructed with existing IT services and new services (e.g., 3G services) without considering common functions between them. Consequently, the study of standardized Service Delivery Platform (SDP) structure and interfaces can efficiently provide the methodology of existing or new services to users and easier environment for development to service providers. Therefore in this paper, we will find solutions about the SDP structure and then describe the implementation of presence service related to the SDP.

1 Introduction

Currently, some services development and their deployment methodologies provided by Service Providers are made up of the form of separated vertical service structures. And these may cause complication, intensiveness of resource and increasing cost of the management. Therefore, we are in face of a platform that can develop, integrate and preserve the horizontal common services to eliminate these complications. This anticipant platform enables the IP Multimedia Subsystem (IMS) structure to provide the control of service consistently and integrate wire and wireless network to meet those demands. The IMS calls for the creation of the SDP that can develop new innovative services based on existing service of the IMS [1].

The SDP simplifies among service interactions through service providers, prevents the overlap of resources through common enablers. Because of these capacities, the SDP can provide new services and reduce processes of service development, complications. Therefore service providers can easily manage services. Also the SDP extends the IMS service layer with containing functions such as third-party service registration, device management, network service, network gateway and so on[2, 3].

Nowadays there isn't a common standard structure about the SDP. Some vendors designate that the SDP is application servers, content solutions and components like Customer Relationship Management (CRM) and Charging. Therefore in this paper, we will suggest structures of the SDP based on the IMS and then show an implementation of presence service using these structures.

2 Composition Elements and Interfaces of the SDP

Figure 1 represents composition elements and interfaces of the SDP in detail. The IMS services are able to access the SDP through SIP AS or OSA-SCS (through Parlay API). Also, the intra-bus that composes the SDP is defined by SOA/Web Service (WS). And SIP interface is defined for the inter-bus of each SDP [4].

To access the IMS network, a module is existed as various protocol gates in a middleware structure. The middleware carries out the necessary protocol conversion between high-level API command and protocol. Supported APIs can be defined as private one or standard ones (e.g., Parlay API, SS7-family SIP, MM7, LIP, SMPP, PAP, SMTP, POP3).

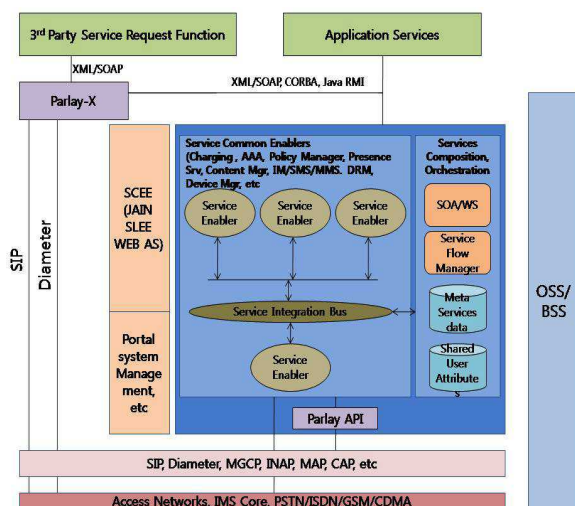


Fig. 1. Composition Elements and Interfaces of the SDP

Service Logic Execution Environment (SLEE), deals with scalability, distribution, configuration of service, is a real-time environment for service providers. The candidate technologies are based on J2EE/.NET, SIP Servlet, JAIN SLEE and Web Service.

Common service enabler supports a basis structure within the service domain. A service developer is able to control and prepare application service that uses the service capabilities provided by common service enablers. Accordingly, service providers don't have to re-create these service functions as stove-pipes. This service enabler includes content management, device management, policy control, logging, subscriber/profile management and so on. Service composition and orchestration is used to create service in a fast way [5].

There is a modeling tool that might compose a desirable service. The communication among services is accomplished as Service Integration Bus (SIB) that is based on Business Process Engineering Language (BPEL). The SIB uses SOA/WS technology as an interface between one service enabler and other service enablers. The Service

Creation & Execution Environment (SCEE) uses a GUI-based tool for making a real-time service bundling. Portal System is used as a third-party management tool. And resource adapter as the SDP service component takes a role of access gateway to the IMS network.

3 An Implementation of Presence Service Based on the IMS

This implementation is based on Jain-sip-presence-proxy API by NIST. In the SDP, presence service attends to common enablers as one of the various common enablers [6,7,8].

Presence Agent (PA) is able to process SIP message (PUBLISH, SUBSCRIBE, NOTIFY). The HSS processes related to authentication, authorization and services via Location Information Request (LIR) and Location Information Answer (LIA).

PUBLISH message is used to upload application services or user information (e.g., status, location) to PA. It doesn't ingenerate an event message but upload its changed information to a server. SUBSCRIBE message is ingenerated specific events, and receives NOTIFY message as a response message. It contains URIs of the servers and users, authentication information, connecting information, expires, created time and home directory of receiving user. NOTIFY message is used to response about specific events (SUBSCRIBE). It includes the URI of the receiving user and contact information, related to authentication information (black list or white list), expires, created time and home directory information. Authentication information of UA can know through verifying initial Filter Criteria (iFC) that indicates whether users are able to use services or not.

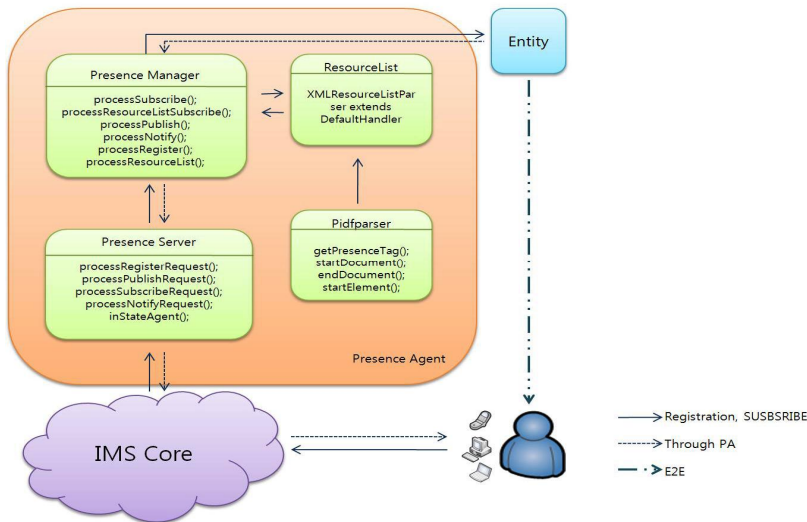


Fig. 2. Structure of PA and the way of E2E and through PA in detail

When entities and users upload their presence information using PUBLISH message via their devices, PA registers its presence information such as online and busy, after authenticating via the HSS. If the user wants to ingenerate specific events using SUBSCRIBE message, PA sends it to suitable entity and it replies to the user as the response of specific events through NOTIFY message and then the entity connects its service to the user.

Figure 2 represents Structure of PA and the way of E2E and through PA in detail. In the way of E2E, PA is used to register presence information of UA and Entity. If PA is concerned in NOTIFY message from entity, PA is able to control QoS, charging, authentication and so on.

When SIP message accesses from S-CSCF, the roles of Presence Server (PS) accomplish to register presence information and upload registering information to *ResourceList*.

Pidfparser defines related to information (e.g., presence, presentity, location and status of device) of users and entities using Presence Information Data Format (PIDF). PIDF is defined in RFC 3863. PIDF is highly extensible, so it is possible to extend the format whenever there is a need to cross beyond the minimal model. Some extensions are being designed aimed at providing a more accurate view of the presence of presentity. PIDF encodes the presence information in an XML document that can be transported, like any other MIME document, in presence publication (PUBLISH transaction) and presence subscription/notification (SUBSCRIBE/NOTIFY transactions) operations [9].

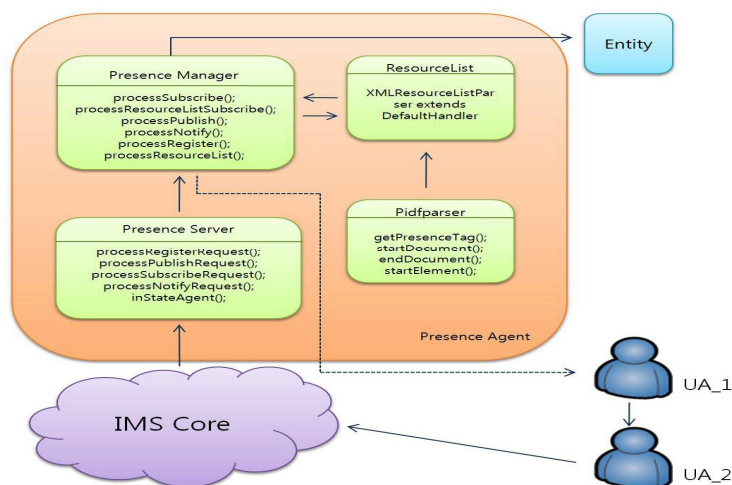


Fig. 3. The way of Changing Status of Devices

ResourceList is implemented possible to store resource information of users and entities. Later on, it will be extended to interact with the HSS for Authentication, Authorization, Accounting (AAA) and Quality of Service (QoS).

If the user that sends SUBSCRIBE message or PUBLISH message for registration in *ResourceList* is existence in *ResourceList*, PM re-uploads partly modified presence information or creates PIDF document about presentity. Presence Manager (PM) makes suitable SIP messages to process SIP messages from PS.

Figure 3 represents that the user device moves from PC1 to PC2 or PC to PDA in order to access entity. In two kinds of scenarios, PM performs de-registration or re-registration through SIP message. In the way of other devices when the user moves from UA_1 to UA_2, UA_2 access to PA using new SUBSCRIBE/PUBLISH message, simultaneously PA sends NOTIFY message in order to de-register UA_1 through comparing SIP header information of UA_1 with UA_2. In this way, the From field of SIP header is the same in UA_1 and UA_2. Otherwise the Contact field of SIP header is different in UA_1 and UA_2. In the way of changing device, PA maintains connection of UA_1 and the status of UA_2 announces to entity like on-line/busy using PUBLISH message.

When 3rd party providers develop contents, they have implemented contents as well as Page 560 of 5 in the past, but if they use PA as common enablers of the SDP, they could easily develop contents without developing PA.

4 Conclusions

Now we have presented the main function and beautiful perspective of the SDP. In AS Layer, the SDP architecture holds the promise of seamless converged voice/data services by various kinds of standards and platforms. The SDP gives the users greater ease of use in multimedia enhanced mobile communications. The SDP plays a role of bridge between established network and new IMS network. By simplifying the interaction among the services, the SDP reasonably provides new innovative services for the market and makes service management problem easier by reducing the complication of service development. Therefore the SDP must combine each service in existing network environment. Implementation of presence service represents a part of the SDP AS. Later on, the presence service model should be much enhanced through extending presence information and its capability. No doubt the IMS/SDP structure will continue on in the future, further strengthening its leadership role, helping customers in their on demand business enablement and the growth, availability, and systems management.

References

1. Cho, J.-H., Lee, J.-O.: SDP and Application Services. KNOM Review (2007)
2. The Future of SDP, Heavy Reading 3(19) (November 2005)
3. Pavlovski, C.J.: Service Delivery Platforms in Practice
4. Cho, J.-H., Lee, J.-O.: Deploying Application Services Using SDP. In: Ata, S., Hong, C.S. (eds.) APNOMS 2007. LNCS, vol. 4773, pp. 575–578. Springer, Heidelberg (2007)
5. Adams, R., Boisseau, M., Bowater, R.: Service Orchestration Implementation Options. In: IBM 2004 (2004)
6. NIST, <http://snad.ncsl.nist.gov/projet/iptel>
7. Jain, S.A.: 1.2 specification (December 2006)
8. RFC 3856 A Presence Event Package for the Session Initiation Protocol (SIP) (August 2004)
9. RFC 3863 Presence Information Data Format (August 2004)

Detection and Handling of TRUNK Congestion in the ATM Network

Charlie Yang, Chuan-Chuen Chang, Chi-Ming Chen, David Lu,
and Monowar Hossain

AT&T, 200 Laurel Ave, Middletown, NJ 07748 USA
{ccy, ccchang, chimingchen, dlu, mhossain}@att.com

Abstract. This paper presents a management structure and methodology for determining a root cause of trunk delay due to trunk congestion events. Based on trunk delay measurements, we conclude that a High Trunk Utilization (HTU) alert can be determined by correlating Constant Bit Rate (CBR) and Variable Bit Rate-non real time (VBR-nrt) trunk delay data. An automated adaptive procedure is developed to prevent trunk congestion before the user is impacted.

1 Introduction

Today with very significant increases in system reliability, users have now come to expect more in the way of “flow through” while “bursting” over the Wide Area Network. In fact, thousands of hours of trouble isolation are expended each year by the Telecom Industry while working congestion issues. Furthermore, network outages do occur due to a variety of reasons with possible lengthy downtimes. Therefore, it has become increasingly important to detect and prevent all problems before the user is impacted [1]. For that reason, we have developed pro-active and adaptive mechanisms [2] to monitor for utilization or over utilization on trunks and respond to any changes in utilization.

2 Management Structure

Figure 1 is a block diagram depicting an exemplary packet network which includes Internet Protocol (IP) Core network, Asynchronous Transfer Mode (ATM) Backbone networks, Access Network, and User Premises to support Digital Subscriber Line (DSL) service.

As shown in Figure 1, the management structure comprises the following capabilities:

- (1) recognizes congested or failed trunk through alarmed event,
- (2) correlates alarmed event, trunk delay, and loss measurement to utilization spike/increase,
- (3) determines an optimization of the trunk or trunks that acquired virtual connections from the congested,
- (4) re-builds the optimal route for trunks that were identified as having utilization increase/spike or connection count increase, and
- (5) auto informs user or appropriate work center through Interactive Voice Response (IVR), email, pager, Instant Message (IM) etc.

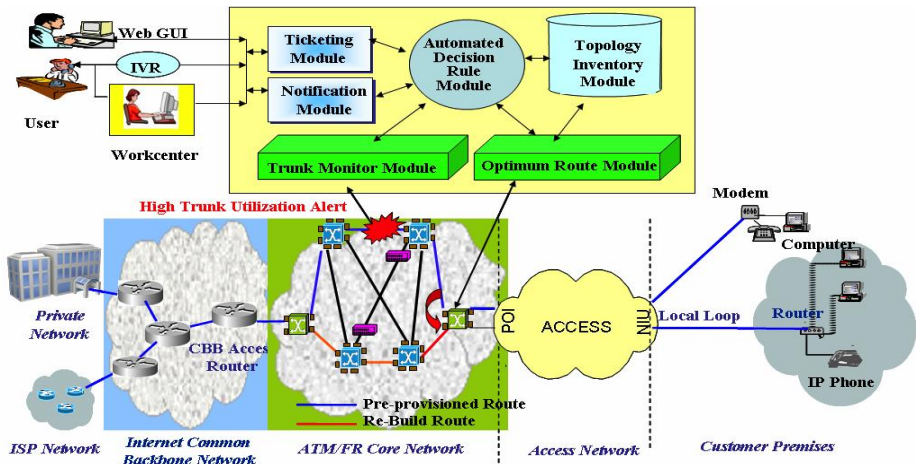


Fig. 1. An Exemplary Packet Network and Management Structure

3 Methodology

In an effort to measure the trunk performance, the round trip cell delay is being collected by the trunk monitoring module. Probes of test cells are sent from one edge of the network to the other edge at a rate of 10 cells per second for one minute. These measurements are repeating every 15 minutes.

Delay can be the result of several different factors [3]. These factors include the speed of the computers involved, the size of the facility connecting the end computer to the ATM, overloaded ATM processes, and poor network routing (too many hops or miles to travel) or congestion on the network. Of these factors, only some of these are within the ATM Service Provider's control (e.g. ATM process, routing and congestion). Others, such as the size of the facility connecting the end computer, have nothing to do with the performance of an ATM backbone.

A HTU alert can be determined based on the combination of CBR and VBR-nrt trunk delay. The high-level of methodology is described as follows:

- (1) Set up a set of selected pre-defined/ preferred route PDP PVCs in Networks
- (2) Use the Probes to measure PVC round trip delay. Delay of a PVC is based on the delay measurement of test cells injected into PVC within the first minute of each 15-minute interval of each hour.
- (3) Calculate the dynamic trunk delay by QoS based on the pair of PVC measurement.
- (4) Determine whether an alert is due to a trunk congestion event or a fiber cuts. For a HTU alert:
- (5) A HTU alert will be generated when the VBR-nrt traffic is impacted by a trunk congestion event.

4 Experiment Result

The following trunk delay measurements were performed and observed:

- (1) As shown in Figure 2, the maximum VBR-nrt round trip delays (RTD) increased from 5 ms to 16 ms when trunk utilization reached to 90%.
- (2) As shown in Figures 3, the maximum and average CBR round trip delays were independent of trunk utilization.
- (3) As shown in Figure 4, when a single fiber cut occurs, the Layer 1 equipment rerouted traffic. Although the trunk was not down and trunk reroute happened, both CBR and VBR-nrt trunk delays increased. Even the trunk is on the protection path, the maximum and average CBR RTDs went up from 7 ms to 31 ms.

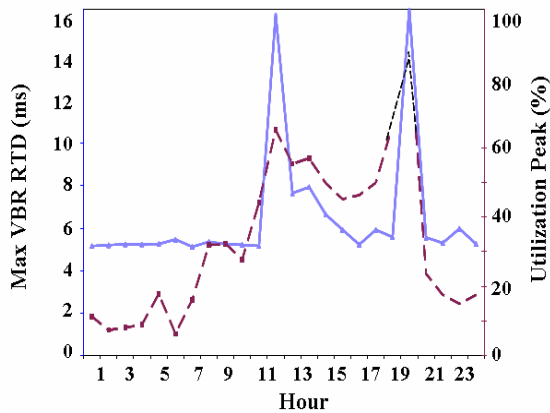


Fig. 2. Maximum VBR-nrt RTD (Solid Line) vs. Max 5 min % Trunk Utilization (Dashed Line)

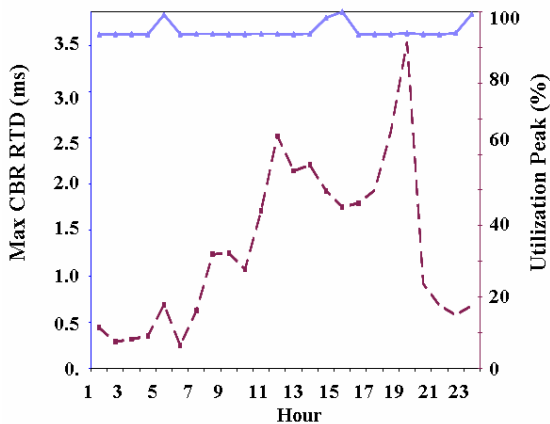


Fig. 3. Maximum CBR RTD (Solid Line) vs. Max 5 min % Trunk Utilization (Dashed Line)

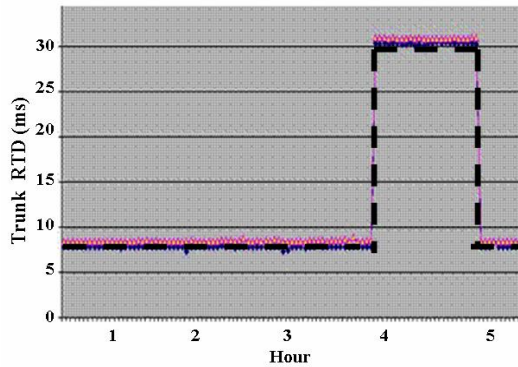


Fig. 4. Average (Dashed line) and Maximum (Solid Line) CBR RTDs When A Fiber Cut Occurs

Based on these measurements, we report that VBR-nrt traffic is impacted by high trunk utilization with high delay and loss, while CBR traffic, as long as not burst, will not be impacted by high trunk utilization. Therefore, we conclude that a HTU alert can be determined by correlating CBR and VBR-nrt trunk delay data. An appropriate threshold of HTU alert should be set by the management structure to detect and prevent trunk congestion.

Acknowledgements. The authors would like to thank Paritosh Bajpay, Mark Hunt, and Michael Lambert for their many helpful suggestions and comments.

References

1. AT&T's Customer Network Management (CNM), <https://fastrakview.com/>
2. Method And Apparatus For Managing Packet Congestion, described in an application for Letters Patent of the United States, filed on June 15, 2007, under U.S. Serial No. 11/763,950
3. Wu, T., Yoshikai, N.: ATM Transport and Network Integrity. Academic Press, London (1997)

Author Index

- Agoulmine, Nazim 146
Ahn, Kwang Seon 539
Ahn, Sengjin 458
Amin, Syed Obaid 483
An, Changqing 256
Arai, Daisuke 41
Ata, Shingo 236

Barreto, Priscila Solís 31
Burnham, Keith J. 21

Caceres, Guillermo
 Horacio Ramirez 523
Cao, Jianting 217
Carvalho, Paulo H.P. de 31
Ceppo, Maximiliano Prestes 31
Chang, Byeong-Yun 297, 556
Chang, Chuan-Chuen 565
Chao, Chi-Shih 61
Chen, Chi-Ming 565
Chen, Ping 227
Chen, Yi-Shian 507
Cheng, Lu 72
Cho, Jae-Hyoung 560
Choi, Byung Goo 177
Choi, Deokjai 399
Choi, Jin-Young 187
Choi, Mi-Jung 156, 474, 531
Choi, Myung Sil 539
Chung, Byung-Deok 556
Chung, Byungdeok 82, 297
Chung, Jinwook 458
Chung, Tai-Myoung 123, 187, 266

Ding, Yongsheng 503
Donnelly, Willie 146

Ejiri, Masayoshi 156

Fan, Wei 466
Fathy, Mahmood 552
Foghlú, Mícheál Ó 146
Fujishima, Daisuke 136

Gao, Tianhan 437
Gkamas, Vasileios 495
Gong, Jian 419

Goto, Kumiko 478
Gu, Rentao 429
Guo, Nan 437

Han, Byung-Jin 123
Han, Seung-Hee 433
Han, Young-Ju 187, 266
Harada, Shigeaki 409
Haribabu, K. 102
Hayashi, Michiaki 317
Heo, Jihyuk 207
Higashi, Kazuhiko 478
Hirata, Shoichi 136
Hirokawa, Yutaka 409
Hong, Choong Seon 177, 207, 462, 483
Hong, Daniel Wonkyu 297, 556
Hong, James Won-Ki 156, 474, 531
Hong, Minhwo 429
Hossain, Monowar 565
Hota, Chittaranjan 102
Huang, Lisheng 511
Huang, Shaojun 256
Huang, Xiaohong 370
Hunter, Mark 487
Hwang, Chan-Kyou 433
Hwang, Junseok 527

Idoue, Akira 41

Jang, Byeong-Yun 535
Ji, Yuefeng 429, 441, 445
Jiang, Dongchen 217
Jin, Yi 217
Ju, Hong-Taek 156
Jung, Kwang-Min 287
Jung, Sung-Min 187
Jung, Young-Chul 276, 344

Kalochristianakis, Michalis 495
Kang, Chul-Hee 287
Kang, Joon-Myung 531
Karimi, Ouldooz Baghban 552
Kashihara, Shuntaro 547
Kato, Minoru 478
Kato, Yuka 543

- Katsuyama, Tsuneo 51
 Kawahara, Ryoichi 409
 Kikuchi, Shinji 51
 Kim, Bom-Soo 433
 Kim, Dae Sun 462
 Kim, Hyeong-Jun 287
 Kim, Igor 344
 Kim, Jin Ho 207
 Kim, Jinsik 82
 Kim, Jong-Myoung 266
 Kim, Kisu 458
 Kim, Kyu-Hwal 535, 556
 Kim, Kyung-Hoe 287
 Kim, Myung-Sup 380, 474
 Kim, Sung-Su 156
 Kim, Sungchun 499
 Kim, Taehoon 112
 Kim, Yonghyoun 458
 Kim, Young-Tak 276, 344
 Kinoshita, Kazuhiko 51
 Kishikawa, Yuki 136
 Kitano, Shin'ichiro 236
 Kobayashi, Kosei 515
 Kong, Lingshan 363
 Koszalka, Leszek 21
 Kwak, Eunjoo 297
 Kwon, Min-kyu 535

 Leaney, John 487
 Lee, Giljae 266
 Lee, Jae-Jin 433
 Lee, Jae-Oh 560
 Lee, Jaesik 499
 Lee, Jangsu 499
 Lee, Jong-Hyouk 123
 Lee, Samson 487
 Lee, Sanghyuc 499
 Lee, Sungju 499
 Lee, Sungwoo 82
 Li, Hui 445
 Li, Xiaofeng 334
 Lin, Zhi-Ting 1, 449
 Liu, Feng 307
 Long, Huihu 72
 Lu, David 565
 Lu, Yueming 441, 445
 Luo, Min 445

 Ma, Hao 227
 Ma, Yan 370
 Majima, Souhei 390, 491

 Mao, Hong-bao 454
 Mamun-Or-Rashid, Md. 462
 Matsumoto, Nobutaka 317
 Matsunaga, Yasuhiko 515
 Matsuura, Hiroshi 11
 Mazoochi, Mojtaba 552
 Meng, Luoming 307
 Miyamoto, Takahiro 317
 Miyazaki, Wataru 51
 Miyoshi, Yu 491
 Mochizuki, Yujiro 478
 Monowar, Muhammad Mostafa 177
 Morita, Naotaka 11
 Murakami, Koso 51
 Mylonas, Georgios 495

 Nguyen, Huy Anh 399
 Nikolettseas, Sotiris 495
 Ning, Zhuo 419
 Niu, Xiaona 246
 Nyu, Takayuki 515

 O'Neill, Tim 487
 Ogaki, Kenichi 547
 Ogishi, Tomohiko 92
 Oimatsu, Toshio 390
 Oka, Ikuo 236
 Oliveira, Rafael Dias 31
 Otani, Tomohiro 547

 Park, Byungchul 474
 Park, Chang-Keun 531
 Park, Jun-Sang 380
 Park, Sukji 82
 Park, Yong Soo 539
 Pozniak-Koszalka, Iwona 21

 Qiao, Xiuquan 334
 Qu, Hua 454
 Qu, Yu-Gui 1, 449

 Rahman, Md. Obaidur 177
 Rolim, Jose 495

 Satake, Takashi 354
 Sato, Yasuhiro 236
 Sekiguchi, Atsuji 51
 Seo, Sin-Seok 531
 Seok, Seung-Joon 287
 Seok, Woojin 266

- Serrat, Joan 146
 Shin, Andrei 527
 Shin, Jung-hwa 112
 Shu, Huaying 466
 Siddiqui, Muhammad Shoaib 483
 Sin, Hongjoong 499
 Strassner, John 146
 Sugahara, Hiroto 515
 Suk, Seung-Hak 535
 Sun, Hongbin 503

 Tahara, Mitsuho 390, 491
 Tak, Sungwoo 112
 Takahashi, Kazuhide 136
 Tamura, Hironao 136
 Tanaka, Hideaki 317
 Tanaka, Yoshiaki 92
 Tateishi, Naoki 390, 491
 Teshigawara, Yoshimi 523
 Tode, Hideki 51
 Tsuchiya, Satoshi 51

 Varvarigos, Emmanouel 495

 Wang, Fang 466
 Wang, Hongxiang 429
 Wang, Hui 256
 Wang, Lili 217
 Wang, Qing-Hua 1, 449
 Wang, Wen-Fong 507
 Wang, Wenyong 511
 Wei, Lihong 363
 Wen, Bo 307
 Won, Young J. 474
 Wu, Jiangxing 246
 Wu, Li 72
 Wu, Lih-Chyau 507

 Xiao, Debao 167
 Xin, Zhanhong 327
 Xu, Hui 167
 Xu, Junjie 327

 Yamamoto, Kimihiro 409
 Yan, Qiang 466
 Yang, Charlie 565
 Yang, Jia 227
 Yang, Jiahai 256, 519
 Yang, Yang 519
 Yang, Young-Moon 82
 Ylä-Jääski, Antti 102
 Yoo, Seunghwan 499
 Yoon, Hyenyoung 527
 Yoshihara, Kiyohito 41
 Yoshiura, Noriaki 470
 You, Tian 334
 Yousefi, Saleh 552
 Yuan, Shuai 217
 Yue, Xin 327

 Zeng, Yongguo 72
 Zhang, Bei 227
 Zhang, Bin 437
 Zhang, Guanqun 519
 Zhang, Jin 246
 Zhang, Zhanyang 197
 Zhanikeev, Marat 92
 Zhao, Bao-Hua 1, 449
 Zhao, Fang 370
 Zhao, Haijun 370
 Zhao, Ji-hong 454
 Zhao, Tong 441
 Zhou, Gang 217
 Zhou, Jingjing 519
 Zhou, Mingtian 511
 Zydek, Dawid 21