

# TUTORIALS<sup>IN</sup> OPERATIONS RESEARCH

2007

## Tutorials in Operations Research

OR Tools and Applications:  
Glimpses of Future Technologies

Theodore Klastorin  
Tutorials Chair and Volume Editor

Paul Gray, Series Editor  
Harvey J. Greenberg, Series Founder

Presented at the INFORMS Annual Meeting, November 4–7, 2007

Copyright ©2007 by the Institute for Operations Research and the  
Management Sciences (INFORMS).

ISBN 13 978-1-877640-22-3

To order this book, contact:

INFORMS  
7240 Parkway Drive, Suite 310  
Hanover, MD 21076 USA  
Phone: (800) 4-INFORMS or (443) 757-3500  
Fax: (443) 757-3515  
E-mail: [informs@informs.org](mailto:informs@informs.org)  
URL: [www.informs.org](http://www.informs.org)

## Table of Contents

Foreword	iv
Preface	vi
Acknowledgments	viii
Chapter 1	
Nested Partitions Optimization	1
<i>Leyuan Shi and Sigurdur Ólafsson</i>	
Chapter 2	
Computational Global Optimization	23
<i>Leon S. Lasdon and János D. Pintér</i>	
Chapter 3	
Coherent Approaches to Risk in Optimization Under Uncertainty	38
<i>R. Tyrrell Rockafellar</i>	
Chapter 4	
Differential Games in Marketing Science	62
<i>Gary M. Erickson</i>	
Chapter 5	
Safe Scheduling	79
<i>Kenneth R. Baker and Dan Trietsch</i>	
Chapter 6	
Community-Based Operations Research	102
<i>Michael P. Johnson and Karen Smilowitz</i>	
Chapter 7	
Generating Robust Project Baseline Schedules	124
<i>Willy Herroelen</i>	
Chapter 8	
Trends in Operations Research and Management Science Education at the Introductory Level	145
<i>Frederick S. Hillier and Mark S. Hillier</i>	
Chapter 9	
Business Engineering: A Practical Approach to Valuing High-Risk, High-Return Projects Using Real Options	157
<i>Scott Mathews and Jim Salmon</i>	
Contributing Authors	176

## Foreword

Tutorials are the lifeblood of our professional society. They help introduce people to fields about which they previously knew little. They stimulate people to examine problems they would not otherwise consider. They help point people to the state of the art and to important unsolved problems. It is no surprise that tutorials are one of the major activities at the INFORMS annual meetings.

The good news this year is that the INFORMS Board of Directors approved distributing a CD of the TutORials volume to every registrant at the Seattle meeting and at the 2008 and 2009 annual meetings. Attendees are urged to take the CD back to their institution and make the contents available to colleagues and students. The printed TutORials book will continue to be available for a small fee.

## History of TutORials

Each year, about 15 tutorials are presented at the INFORMS meeting. Although the attendance at tutorial sessions is among the largest of all sessions—numbers around 200 in a single session are common—until three years ago, their important content was lost to the many INFORMS members who could not attend either the tutorial sessions or the annual meeting itself. Clearly, INFORMS was underusing one of its treasures.

In 2003, Harvey Greenberg of the University of Colorado at Denver, founding editor of the *INFORMS Journal on Computing* and well known for his many contributions to OR scholarship and professional service, was appointed the tutorials chair for the Denver National Meeting. He recognized the problem that tutorials were only available for listening once to those who attended them. The result was a lack of institutional memory. He decided to do something about it. He organized the *TutORials in Operations Research* series of books. His idea was that a selection of the tutorials offered at each annual meeting would be prepared as chapters in an edited volume widely available through individual and library purchase. To increase their circulation, the books would also be available at the INFORMS fall annual meetings.

Harvey edited the Tutorials book for the Denver INFORMS meeting in 2004, which was published by Springer. In 2005, Frederick H. Murphy, then Vice President, Publications, of INFORMS, working closely with Harvey, convinced the INFORMS Board of Directors to bring the *TutORials* series under the umbrella of our society. Harvey was appointed Series Editor. He, in turn, asked J. Cole Smith of the University of Florida, Tutorials Chair of the San Francisco Annual Meeting, to serve as editor of the 2005 volume, the first to be sponsored by INFORMS. In doing so, Harvey initiated the policy that the invited Tutorials chair also serve as the volume editor. As the result of a suggestion by Richard C. Larson, 2005 President of INFORMS, a CD version of the volume was also created. In mid-2005, Harvey Greenberg, nearing retirement, asked to relinquish the series editorship. Paul Gray was appointed to replace him.

Last year, the Pittsburgh meeting Chair, Michael Trick, appointed three Tutorials Co-chairs—Michael P. Johnson and Nicola Secomandi of Carnegie Mellon University and Bryan Norman of the University of Pittsburgh—who served as Co-Editors of the 2006 volume. This year's volume editor is Theodore Klastorin, distinguished professor at the University of Washington, who is the tutorials chair for the Seattle meeting. He assembled nine tutorials

for this volume that, as in previous years, cover a broad range of fields within OR. These tutorials include

- Nested participation optimization
- Computational global optimization
- Risk in optimization under uncertainty
- Differential games in marketing science
- Safe scheduling
- Community-based operations research
- Project management
- Using options theory to assess projects
- Trends in OR and MS education at the introductory level

The authors come from major universities and a world-known company. They include (alphabetically) The Boeing Company, Carnegie Mellon University, Dartmouth College, Catholic University of Leuven (Belgium), Northwestern University, Stanford University, University of Texas, University of Washington, and University of Wisconsin.

On behalf of the INFORMS membership, I thank the volume editor for creating this year's exciting tutorial series and doing the enormous amount of work required to create this volume. INFORMS is also indebted to the authors who contributed the nine chapters.

The tutorial series also benefits from the work of its Advisory Committee, consisting of Harvey J. Greenberg (University of Colorado at Denver and Health Sciences Center), Frederick S. Hillier (Stanford University), Michael P. Johnson (Carnegie Mellon University), J. Cole Smith (University of Florida), and David Woodruff (University of California, Davis). Finally, an important thank you to Miranda Walker, Kate Lawless, Patricia Shaffer (Director of Publications), and the members of the publications staff at the INFORMS office for the physical preparation of this volume and its publication in a timely manner.

PAUL GRAY  
Claremont Graduate University  
Claremont, California

## Preface

Welcome to the *TutORials in Operations Research* 2007, subtitled “OR Tools and Applications: Glimpses of Future Technologies”; this is the fourth published volume in the series that was started by Harvey J. Greenberg in 2004. Like its predecessors, the tutorials in this book, as well as eight others, will be presented at the 2007 INFORMS annual meeting in Seattle, Washington. They all represent both the breadth and depth of methodologies and applications that define operations research’s (OR) varied and significant contributions. This year, we are fortunate to have a group of tutorials presented by mostly senior INFORMS members who have extensive experience in advancing their respective topics as well as applying and presenting their material.

OR has been applied to numerous problems in the nonprofit sector with considerable success. For example, the potential and challenges of work in this area are illustrated by the tutorial by Michael P. Johnson (“Community-Based Operations Research”). In his chapter (co-authored with Karen Smilowitz), Johnson argues that political and social forces cause public sector problems to be “messier” and more complex than problems in the private sector. Johnson and Smilowitz illustrate their points with case studies in the areas of food security and affordable housing.

In other presentations, Kenneth R. Baker presents a tutorial on “safe scheduling”. In his chapter (co-authored with Dan Trietsch), Baker presents a review of scheduling theory and describes how results from deterministic scheduling theory can be extended to stochastic problems to accommodate safety times explicitly. In a related tutorial, Willy Herroelen discusses the issue of generating project planning schedules that are protected from random disruptive events and unanticipated delays (“Generating Robust Project Baseline Schedules”). Extending the topic of planning under the threat of possible disruptive events, R. Tyrrell Rockafellar’s tutorial (“Coherent Approaches to Risk in Optimization Under Uncertainty”), explores various strategies for optimizing stochastic planning problems and the mathematical implications of these strategies, as well as recently developed methodologies that avoid some of the problems (e.g., nonconvexity) inherent in previously used approaches.

Extending optimization related topics, Leon S. Lasdon presents a tutorial on solving global optimization problems (“Computational Global Optimization”). In his chapter (co-authored with János D. Pintér), Lasdon discusses various strategies that have proven effective for finding global optima in nonlinear models that have multiple local and global optima. Lasdon and Pintér review previously suggested solution approaches and present current problem areas and research opportunities. In a related tutorial, “Nested Partitions Optimization,” Leyuan Shi, presents the nested partitions (NP) method for solving large-scale discrete optimization problems. In her chapter (co-authored with Sigurdur Ólafsson), she discusses how to implement the NP method, how it relates to other exact and heuristic methods, and illustrates the applicability of the NP method to various problems in supply chain management, health case delivery, and data mining.

In the years since the first OR application was published by A. Charnes in 1956, the profession has matured in both the development and application of OR methodologies and tools. In his tutorial, Frederick S. Hillier (with co-author Mark S. Hillier) discuss the trends in OR and management science education over the past forty years. Hillier’s tutorial discusses changes that have occurred and explore what changes might be expected in the future.

Exploring the interface between OR methodologies and related areas, Gary M. Erickson will present a tutorial on “Differential Games in Marketing Science”. In his chapter, Erickson

presents an overview of differential games as applied in marketing science and advertising. Erickson's tutorial should be of interest to Operations Management researchers, among others, who are interested in expanding their work to include marketing issues. Erickson explores the challenges of using differential games and will discuss a number of applications and numerical examples.

In the final chapter, Scott Mathews (with Jim Salmon) of The Boeing Company discusses an important application of OR/finance methodologies to the problem of planning and evaluating risky proposed projects. Their approach (termed "business engineering") combines concepts from real options with Monte Carlo simulation and demonstrates how their multidisciplinary methodology can provide firms with a tool that can help them evaluate and structure high-benefit projects while minimizing risks.

These 2007 *TutORials*' chapters illustrate the contributions that derive from an interdisciplinary field that lies at the intersection of economics, engineering, computer science, mathematics, probability and statistics, and psychology. The tutorials demonstrate how OR can bring synergy, insights, and solutions to complex problems. It is the goal of the 2007 INFORMS tutorials committee to represent a breadth and width of current problem areas and tools that define our profession. We hope that all the tutorials presented will stimulate additional interest, research, and application in the many areas where OR has much to contribute.

TED KLASTORIN  
University of Washington  
Seattle, Washington

## Acknowledgments

Many people have assisted with the substantial effort to organize the 2007 *Tutorials in Operations Research*. My great thanks to Paul Gray who serves as the series editor but has provided much more . . . serving as a mentor, advisor, reviewer, and project planner. INFORMS staff members Miranda Walker, Patricia Shaffer, and Kate Lawless have worked tirelessly to prepare this volume in a timely manner despite my (unintentional) efforts to derail their efforts. Thanks also to Seattle INFORMS Chair Zelda Zabinsky who continually offered her constant support and optimism even when conditions dictated otherwise. My deepest gratitude to the anonymous reviewers who assisted with the review and editing of these papers under very tight deadlines; I am most grateful for their counsel and hard work. And, last but certainly not least, I thank my colleagues who have contributed to this volume as well as everyone who has agreed to present a tutorial at the 2007 INFORMS meeting in Seattle.

TED KLASTORIN  
University of Washington  
Seattle, Washington



# Nested Partitions Optimization

*Leyuan Shi*

Industrial and Systems Engineering Department, University of Wisconsin–Madison, Madison, Wisconsin 53706, leyuan@ie.engr.wisc.edu

*Sigurdur Ólafsson*

Industrial and Manufacturing Systems Engineering Department, Iowa State University, Ames, Iowa 50011, olafsson@iastate.edu

**Abstract** We introduce the nested partitions (NP) method for solving large-scale discrete optimization problems. As such problems are common in many practical applications the NP method has been found useful in diverse application areas. It has for example been applied to classic combinatorial optimization problems, such as the traveling-salesman problem and production-scheduling problems, as well as more recent applications in data mining and radiation therapy. The tutorial discusses the basic idea of the NP method, shows in some detail how it should be implemented, presents the basic convergence properties of the method, and discusses several successful implementations in diverse application areas.

**Keywords** discrete optimization; metaheuristics; combinatorial optimization; mixed integer programming

---

## 1. Introduction

This tutorial introduces the *nested partitions* (NP) method. The NP method is a powerful optimization method that has been found to be very effective for solving large-scale discrete optimization problems. Such problems are common in many practical applications and the NP method is hence useful in diverse application areas. It can be applied to both operational and planning problems and has been demonstrated to effectively solve complex problems in both manufacturing and service industries.

The NP method was first introduced by Shi and Ólafsson (1997) and its basic properties for discrete optimization were established in Shi and Ólafsson [6]. It has been successfully applied to many classic combinatorial optimization problems, such as the traveling-salesman problem (Shi et al. [8]), and production-scheduling problems (Ólafsson and Shi [6]), as well as more recent applications in data mining (Ólafsson and Yang [4]) and radiation therapy (D’Souza et al. [1]). For a complete treatment of the NP method, we refer the reader to Shi and Ólafsson [7].

The tutorial discusses the basic idea of the NP method, shows in some detail how it should be implemented, presents the basic convergence properties of the method, and discusses several successful implementations in diverse application areas. We start by defining the domain for which the NP method is the most applicable, namely large-scale discrete optimization problems.

## 2. Discrete Optimization

The NP method is particularly well suited for complex large-scale discrete optimization problems where traditional methods experience difficulty. It is, however, very broadly appli-

cable and can be used to solve any optimization problem that can be stated mathematically in the following generic form:

$$\min_{x \in X} f(x), \quad (1)$$

where the solution space or feasible region  $X$  is either a discrete or bounded set of feasible solutions.

An important special case of problem that can be effectively addressed using the NP method are *mixed integer programs* (MIP). For such problems there may be one set of discrete variables and one set of continuous variables and the objective function and constraints are both linear. A general MIP can be stated as follows (Wolsey [9]):

$$z_{\text{MIP}} = \min_{x, y \in X} c^1 x + c^2 y, \quad (2)$$

where  $X = \{x \in \mathbb{Z}_+^n, y \in \mathbb{R}^n: A^1 x + A^2 y \leq b\}$  and we use  $z_{\text{MIP}}$  to denote any linear objective function, that is,  $z_{\text{MIP}} = f(x) = cx$ . Although some large-scale MIPs can be solved efficiently using exact mathematical programming methods, complex applications often give rise to MIPs where exact solutions can only be found for relatively small problems. When dealing with such complex large-scale problems the NP method provides an attractive alternative. However, even in such cases it may be possible to take advantage of exact mathematical programming methods by incorporating them into the NP framework. The NP method therefore provides a framework for combining the complimentary benefits of two optimization approaches that have traditionally been studied separately, namely mathematical programming and metaheuristics.

Another important class of problems are *combinatorial optimization problems* (COP) where the feasible region is finite but its size typically grows exponentially in the input parameters of the problem. A general COP can be stated as follows:

$$\min_{x \in X} f(x), \quad (3)$$

where  $|X| < \infty$ , but the objective function  $f: X \rightarrow \mathbf{R}$  may be a complex nonlinear function. Sometimes it may have no analytic expression and must be evaluated through a model, such as a simulation model, a data-mining model, or other application-dependent models. One important advantage of the NP method is that it is effective for optimization when  $f$  is known analytically (*deterministic optimization*), when it is noisy (*stochastic optimization*), or even when it must be evaluated using an external process.

### 3. Methodology

The NP method is best viewed as a metaheuristic framework, and it has similarities to branching methods in that it creates partitions of the feasible region like branch-and-bound does. However, it also has some unique features that make it well suited for very hard large-scale optimization problems.

Metaheuristics have emerged as the most widely used approach for solving difficult large-scale combinatorial optimization problems (Gendreau and Potvin [2]). A metaheuristic provides a framework to guide application-specific heuristics, such as a greedy local search, by restricting which solution or set of solutions should or can be visited next. For example, the tabu search metaheuristic disallows certain moves that might otherwise be appealing by forbidding (i.e., making tabu) the reverse of recent moves. At the same time it always forces the search to take the best nontabu move, which enables the search to escape local optima. Similar to tabu search, most metaheuristics guide the search from solution to solution or possibly from a set of solutions to another set of solutions. In contrast, the NP method guides the search by determining where to concentrate the search effort. Any optimization

method, such as an application-specific local search, other general-purpose heuristic, or a mathematical-programming method, can then be integrated within this framework.

The development of metaheuristics and other heuristic search methods have been made largely in isolation from the recent advancements in mathematical-programming methods for solving large-scale discrete problems. It is a very important and a novel characteristic of the NP method that it provides a natural metaheuristic framework for combining the use of heuristics and mathematical programming, and that it takes advantage of their complementary nature. Indeed, as far as we know, the NP method is the first systematic search method that enables users to simultaneously realize the full benefits of incorporating lower bounds through various mathematical-programming methods and using any domain knowledge or heuristic search method for generating good feasible solutions. It is this flexibility that makes the NP method so effective for practical problems.

To concentrate the search effort, the NP method employs a decomposition approach similar to that of branch-and-bound. Specifically, in each step the method partitions the space  $X$  of feasible solutions into the *most-promising region* and the *complimentary region*, namely the set of solutions not contained in the most-promising region. The most-promising region is then partitioned further into subregions. The partitioning can be done exactly as branching for a branch-and-bound algorithm, but instead of focusing on obtaining lower bounds and comparing those bounds to a single primal feasible solution, the NP methods focuses on generating primal feasible solution from each of the subregions and the complimentary region. This results in an upper bound on the performance of each of these regions. The region with the best feasible solution is judged the most promising and the search is focused accordingly. A best upper bound does not guarantee that the corresponding subset contains the optimal solution, but because the NP method also finds primal feasible solutions for the complimentary region it is able to recover from incorrect moves. Specifically, if the best solution is found in one of the subregions this becomes the new most-promising region, where if it is in the complimentary region the NP method backtracks. This focus on generating primal feasible solutions and the global perspective it achieves through backtracking are distinguishing features of the NP method, which set it apart from similar branching methods.

Unlike exact optimization methods such as branch-and-bound the NP method does not guarantee that the correct region is selected in each move of the algorithm. Incorrect moves can be corrected through backtracking, but for the method to be both effective and efficient, the correct move must be made frequently. How this is accomplished depends on how the feasible solutions are generated.

In what we refer to as the *pure NP method*, feasible solutions are generated using simple uniform-random sampling. To increase the probability of making the correct move the number of samples should be increased. A purely uniform-random sampling is rarely efficient, however, and the strength of the NP method is that it can incorporate application-specific methods for generating feasible solutions. In particular, for practical applications domain knowledge can often be utilized to very effectively generate good feasible solutions. We call such implementations *knowledge-based NP methods*. We will also see examples of what we refer to as *hybrid NP methods*, where feasible solutions are generated using either general heuristic methods such as greedy local search, genetic algorithm, or tabu search, or using mathematical-programming methods. If this is done effectively, incorporating such methods into the NP framework makes it more likely that the correct move is made and hence makes the NP method more efficient. Indeed, such hybrid and knowledge-based implementations are often an order of magnitude more efficient than uniform-random sampling.

In addition to the method for generating feasible solutions, the probability of making the correct move depends heavily on the partitioning approach. A generic method for partitioning is usually straightforward to implement but by taking advantage of special structure and incorporating this into intelligent partitioning the efficiency of the NP method may be

improved by an order of magnitude. The strength of the NP method is indeed in this flexibility. Special structure, local search, any heuristic search, and mathematical programming can all be incorporated into the NP framework to develop optimization algorithms that are more effective in solving large-scale optimization problems than when these methods are used alone.

## 4. Application Examples

Here we introduce three application examples that illustrate the type of optimization problems for which the NP method is particularly effective. For each application the optimization problem has a complicating aspect that makes it difficult for traditional optimization methods. For the first of these problems, resource-constrained project scheduling, the primary difficulty is in a set of complicating constraints. For the second problem, the feature-selection problem, the difficulty lies in a complex objective function. The third problem, radiation-treatment planning, has both difficult to satisfy constraints and a complex objective function that cannot be evaluated through an analytical expression. Each of the three problems can be solved effectively by the NP method by incorporating our understanding of the application into the framework.

### 4.1. Resource-Constrained Project Scheduling

Planning and scheduling problems arise as critical challenges in many manufacturing and service applications. One such problem is the resource-constrained project scheduling problem that can be described as follows. A project consists of a set of tasks to be performed and given precedence requirements between some of the tasks. The project-scheduling problem involves finding the starting time of each task so that the overall completion time of the project is minimized. It is well-known that this problem can be solved efficiently by using what is called the critical-path method that uses forward recursion to find the earliest possible completion time for each task. The completion time of the last task defines the makespan or the completion time of the entire project.

Now assume that one or more resources are required to complete each task. The resources are limited so if a set of tasks requires more than the available resources they cannot be performed concurrently. The problem now becomes NP-hard and cannot be solved efficiently to optimality using any traditional methods. To state the problem we need the following notation:

$V$  = Set of all tasks

$E$  = Set of precedence constraints

$p_i$  = Processing time of task  $i \in V$

$R$  = Set of resources

$R_k$  = Available resources of type  $k \in R$

$r_{ik}$  = Resources of type  $k$  required by task  $i$ .

The decision variables are the starting times for each task,

$$x_i = \text{Starting time of task } i \in V. \quad (4)$$

Finally, for notational convenience we define the set of tasks processed at time  $t$  as

$$V(t) = \{i: x_i \leq t \leq x_i + p_i\}.$$

With this notation, we now formulate the resource-constrained project-scheduling problem mathematically as follows:

$$\min \max_{i \in V} x_i + p_i \quad (5)$$

$$x_i + p_i \leq x_j, \quad \forall (i, j) \in E \quad (6)$$

$$\sum_{i \in V(t)} r_{ik} \leq R_k, \quad \forall k \in R, t \in \mathbf{Z}_+^1 \quad (7)$$

$$x_i \in \mathbf{Z}_+^1.$$

Here the precedence constraints (6) are easy, whereas the resource constraints (7) are hard. By this we mean that if the constraints (7) are dropped then the problem becomes easy to solve. Such problems, where complicating constraints transform the problem from easy to very hard, are common in large-scale optimization. Indeed the classic job-shop-scheduling problem can be viewed as a special case of the resource-constrained project-scheduling problem where the machines are the resources. Without the machine availability constraints the job-shop-scheduling problem reduces to a simple project-scheduling problem. Other well-known combinatorial optimization problems have similar properties. For example, without the subset elimination constraints the classic traveling-salesman problem (TSP) reduces to a simple assignment problem that can be solved efficiently.

The flexibility of the NP method allows us to address such problems effectively by taking advantage of special structure when generating feasible solutions. It is important to note that it is very easy to use sampling to generate feasible solutions that satisfy very complicated constraints. Therefore, when faced with a problem with complicating constraints we want to use random sampling to generate partial feasible solutions that resolve the difficult part of the problem and then completed the solution using the appropriate efficient optimization method.

For example, when generating a feasible solution for the resource-constrained project-scheduling problem, the resource allocation should be generated using random sampling and the solution can then be completed by applying the critical-path method to determine the starting times for each task. This requires reformulating the problem so that the resource and precedence constraints can be separated, but such a reformulation is rather easily achieved by noting that the resource constraints can be resolved by determining a sequence between the tasks that require the same resource(s) at the same time. Once this sequence is determined then the sequence can be added as precedence constraints and the remaining solution generated using the critical path method. Feasible solutions can therefore be generated in the NP method by first randomly sampling a sequence to resolve resource conflicts and then applying the critical-path method. Both procedures are very fast so complete sample solutions can be generated rapidly.

We also note that constraints that are difficult for optimization methods such as mathematical programming are sometimes very easily addressed in practice by incorporating domain knowledge. For example, a domain expert may easily be able to specify priorities among tasks requiring the same resource(s) in the resource-constrained project-scheduling problem. The domain expert can therefore, perhaps with some assistance from an interactive decision support system, specify some priority rules to convert a very complex problem into one that is easy to solve. The NP method can effectively incorporate such domain knowledge into the optimization framework by using the priority rules when generating feasible solutions. This is particularly effective because the domain expert would not need to specify priority rules to resolve all resource conflicts. Rather, any available priority rule or other domain knowledge can be incorporated to guide the sampling.

The same structure can be used to partition intelligently. Instead of partitioning directly using the decision variables (4), we note that it is sufficient to partition to resolve the resource conflicts. Once those are resolved then the problem is solved. This approach is applicable to any problem that can be decomposed in a similar manner.

## 4.2. Feature Selection

Knowledge discovery and data mining is a relatively new field that has experienced rapid growth due to its ability to extract meaningful knowledge from very large databases. One

of the problems that must usually be solved as part of practical data mining is the feature selection problem, which involves selecting a good subset of variables to be used by subsequent inductive data-mining algorithms. The problem of selecting a best subset of variables is well-known in statistical literature as well as in machine learning. The recent explosion of interest in data mining for addressing various business problems has led to a renewed interest in the feature selection problem. From an optimization point of view, feature selection can clearly be formulated as a combinatorial optimization problem (COP), where binary decision variables determine if a feature (variable) is included or excluded. The solution space can therefore be stated very simply as all permutation of binary vector of length  $n$ , where  $n$  is the number of variables. The size of this feasible region is  $2^n$  so it experiences exponential growth, but typically there are no additional constraints to complicate its structure.

On the other hand, there is no consensus objective function that measures the quality of a feature or a set of features. Tens of alternatives have been proposed in the literature, including both functions that measure the quality of individual features and functions that measure the quality of a set of features but no single measure is satisfactory in all cases. However, the ultimate measure is: Does it work? In other words, when the selected features are used for learning does it result in a good model being induced? The most effective feature-selection approach in terms of solution quality is therefore the wrapper approach, where the quality of a set of features is evaluated by applying a learning algorithm to the set and evaluating its performance. Specifically, an inductive learning algorithm, such as decision tree induction, support vector machines, or neural networks are applied to training data containing only the selected features. The performance of the induced model is evaluated and this performance is used to measure the quality of the feature subset. This objective function is not only nonlinear, but since a new model must be induced for every feature subset it is very expensive to evaluate.

Mathematically, the feature selection can be stated as the following COP:

$$\min_{x \in \{0,1\}^n} f(x), \quad (8)$$

that is,  $X = \{0,1\}^n$ . Feature selection is therefore a very hard combinatorial optimization problem not because of the complexity of the feasible region, although it does grow exponentially, but due to the complexity of an objective function that is very expensive to evaluate. However, this is also an example where application-specific heuristics can be effectively exploited by the NP method.

Significant research has been devoted to methods for measuring the quality of features. This includes information-theoretic methods such as using Shannon's entropy (Shannon [5]) to measure the amount of information contained in each feature: The more information, the more valuable the feature. The entropy is measured for each feature individually and it can hence be used as a very fast local search or a greedy heuristic, where the features with the highest information gain are added one at a time. Although such a purely entropy-based feature selection will rarely lead to satisfactory results, the NP method can exploit this by using the entropy measure to define an intelligent partitioning.

We let  $X(k) \subseteq X$  denote the most-promising region in the  $k$ th iteration and partition the set into two disjoint subsets (note that  $X(0) = X$ ):

$$X_1(k) = \{x \in X(k): x_i = 1\}, \quad (9)$$

$$X_2(k) = \{x \in X(k): x_i = 0\}. \quad (10)$$

Hence, a partition is defined by a sequence of features  $x_1, x_2, \dots, x_n$ , which determines the order in which the features are either included ( $x_i = 1$ ) or excluded ( $x_i = 0$ ).

We calculate the information gain  $Gain(i)$  of feature  $i$ , which is the expected reduction in entropy that would occur if we knew the value of feature  $i$ , that is,

$$Gain(i) = I - E(i), \quad (11)$$

where  $I$  is the expected information that is needed to classify a given instance and  $E(i)$  is the entropy of each feature. The maximum information gain, or equivalently the minimum entropy, determines a ranking of the features. Thus, we select

$$\begin{aligned} i_1 &= \arg \min_{i \in \{1, 2, \dots, n\}} E(i), \\ i_2 &= \arg \min_{i \in \{1, 2, \dots, n\} \setminus \{i_1\}} E(i), \\ &\vdots \\ i_n &= \arg \min_{i \in \{1, 2, \dots, n\} \setminus \{i_1, \dots, i_{n-1}\}} E(i). \end{aligned}$$

The feature order  $i_1, i_2, \dots, i_n$  defines an intelligent partition for the NP method and this has been found to be an order of magnitude more efficient than an average arbitrary partitioning (Ólafsson [3]). We can use a similar idea to generate feasible solutions from each region using a sampling strategy that is biased toward the inclusion of features with high information gain. A very fast greedy heuristic can thus greatly increase the efficiency of the NP method and result in much higher quality solutions than the greedy heuristic is able to achieve on its own.

### 4.3. Radiation-Treatment Planning

Health care delivery is an area where optimization techniques have been used increasingly in recent years; e.g., Intensity-Modulated Radiation Therapy (IMRT) is a recently developed complex technology for radiation-treatment planning. It employs a multileaf collimator to shape the beam and to control, or modulate, the amount of radiation that is delivered from each of the delivery directions (relative to the patient). The planning of IMRT is challenging because it needs to achieve the treatment goal while incurring the minimum possible damage to other organs. Because of its complexity the treatment-planning problem is generally divided into several subproblems. The first of these is termed the *beam angle selection* (BAS) problem. In essence, BAS requires the determination of roughly four to nine angles from 360 possible angles subject to various spacing and opposition constraints.

When designing an optimal IMRT plan, the clinician manually selects the beam angles from which radiation is delivered to the patient. The planning process proceeds as follows: a dosimetrist selects a collection of angles and waits ten to thirty minutes while a dose pattern is calculated. In practice the resulting treatment is for medical reasons likely to be unacceptable, so the angles and dose constraints are adjusted, and the process repeats. Finding a suitable collection of angles often takes several hours. The goal of using optimization methods to identify quality angles is to provide a better decision support system to replace the tedious repetitive process just described. An integer programming model of the problem contains a large number of binary variables and the objective value of a feasible point is evaluated by solving a large, continuous optimization problem. For example, in selecting five to ten angles, there are between  $4.9^{10}$  and  $8.9 \times 10^{19}$  subsets of  $0, 1, 2, \dots, 359$ .

The BAS problem is complicated by both an objective function with no analytical expression and by constraints that are hard to satisfy. In the end an IMRT plan is either acceptable or not, and the considerations for determining acceptability are too complex for a simple analytical model. Thus, the acceptability and hence the objective function value for each plan must be evaluated by a qualified physician. This makes evaluating the objective not only expensive in terms of time and effort, but also introduces noise into the objective function because two physicians may not agree on the acceptability of a particular plan. The constraints of the BAS problem are also complicated because each beam angle will result in radiation of organs that are not the target of the treatment. There are, therefore, two

types of constraints: The target should receive a minimum radiation and other organs should receive no more than some maximum radiation. Because these bounds need to be specified tightly the constraints are hard to satisfy.

The BAS problem illustrates how mathematical programming can be effectively incorporated into the NP framework. Because the evaluation of even a single IMRT plan must be done by an expert and is hence both time consuming and expensive, it is imperative to impose a good structure on the search space that reduces the number of feasible solutions that need to be generated. This can be accomplished through an intelligent partitioning, and specifically by computing the optimal solution of an integer program with a much simplified objective function (D'Souza et al. [1]). The output of the integer program (IP) then serves to define an intelligent partitioning. For example, suppose a good angle set ( $50^\circ, 80^\circ, 110^\circ, 250^\circ, 280^\circ, 310^\circ, 350^\circ$ ) is found by solving the IP. We can then partition on the first angle in the set, which is  $50^\circ$  in this example. Then one subregion includes angle  $50^\circ$ , the other excludes excluding  $50^\circ$ .

## 5. Implementing the NP Method

The basic implementation of the NP method is very simple, and it is indeed this simplicity that gives it the flexibility to effectively incorporate application-specific structure and methods while providing a framework that guides the search and enables meaningful convergence analysis. Specifically, recall that there are four components of the NP method: partitioning, generating feasible solutions, calculating the promising index, and backtracking. In the next four sections we discuss the implementation of each step in more details.

### 5.1. Partitioning

The partitioning is of paramount importance to the efficiency of the NP method because the selected partition imposes a structure on the feasible region. When the partitioning is done in such a way that good solutions are clustered together then those subsets tend to be selected by the algorithm with relatively little effort. On the other end of the spectrum, if the optimal solution is surrounded by solutions of poor quality it is unlikely that the algorithm will move quickly toward those subsets. For some problems a simple partition may automatically achieve the clustering of a good solution but for most practical applications more effort is needed. We will see how it is possible to partition effectively by focusing on the most difficult decisions and how both heuristics and mathematical programming can be applied to find partitions that improve the efficiency of the algorithm. We refer to such partitions as *intelligent partitioning* to distinguish it from *generic partitioning* that partitions the feasible region without considering domain knowledge, the objective function, or any other special structure.

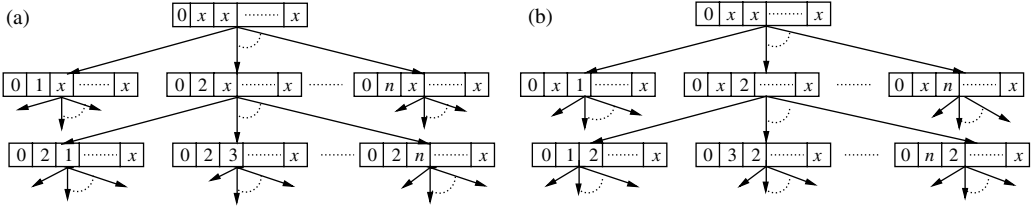
**5.1.1. A Generic Partitioning Method.** We illustrate a generic partitioning through the traveling-salesman problem (TSP). This classic COP can be described as follows. Imagine a traveling salesman who must visit a set of cities. The objective is to minimize the distance traveled, while the constraints assure that each city is visited exactly once (assignment constraints) and that the selected sequence of cities forms a connected tour (subset elimination constraints). Without the subset elimination constraints the TSP reduces to simple assignment problem, whereas with these constraints it is a NP-hard problem, which implies that it is unlikely that a polynomial time algorithm exists for its solution.

Assume that there are  $n + 1$  cities. For a generic partitioning method arbitrarily choose city 0 as the starting point and label the other cities as  $1, 2, 3, \dots, n$ . The feasible region becomes all permutations of  $\{1, 2, 3, \dots, n - 1\}$ ,

$$X = \{x \in Z_+^n: 1 \leq x_i \leq n, x_i \neq x_j \text{ if } i \neq j\}.$$



FIGURE 1. Two generic partitions.



First, partition the feasible region into  $n$  regions by fixing the first city on the tour to be one of  $1, 2, \dots, n$ . Partition each such subregion further into  $n - 1$  regions by fixing the second city as any of the remaining  $n - 1$  cities on the tour. This procedure can be repeated until all the cities on the tour are fixed and the maximum depth is reached. In this way the subregions at maximum depth contain only a single solution (tour). Figure 1(a) illustrates this approach. Clearly there are many such partitions. For example, when we choose city 0 as the starting point, instead of fixing the first city on the tour, we fix any  $i$ th city on the tour to be one of cities  $1, 2, \dots, n$  (see Figure 1(b)). This partition provides a completely different set of subregions, that is, the set  $\Sigma$  of valid regions will be different than before.

This is a generic partition because it does not take advantage of any special structure of the TSP and it does not take the objective function into account. It simply partitions the feasible region without considering the performance of the solutions in each region of the partition. It is intuitively appealing that a more efficient implementation of the NP method could be achieved if the objective function was considered in the partitioning to assure that good solutions are clustered together.

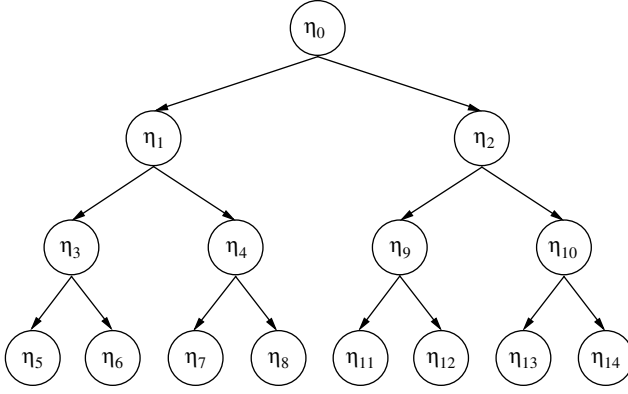
**5.1.2. Intelligent Partitioning for TSP.** The generic partitioning does not consider the objective function when partitioning the feasible region. This may lead to difficulties in distinguishing between regions and consequently the algorithm may not efficiently find where to concentrate the computational effort. If the NP method is applied using the above partitioning, it may backtrack frequently and not settle down in a particular region. On the other hand, the NP method is likely to perform more efficiently if good solutions tend to be clustered together for a given partitioning. To impose such structure, consider the following partitioning scheme through a simple example.

**Example 1.** Assume  $n = 5$  cities are defined by the undirected graph in Figure 2. As an initialization procedure store the edges in an adjacency list and sort each of the linked lists that are connected to the cities (see the following table). For example, in the following adjacency list, the first row provides a linked list for city  $A$ , that is  $E$  is the city closest to  $A$ ,  $C$  is the city second closest to  $A$ ,  $B$  is the city next closest to  $A$ , and  $D$  is the city least close to  $A$ .

City	Closest two	Next two
$A \rightarrow$	$E \rightarrow C \rightarrow$	$B \rightarrow D$
$B \rightarrow$	$C \rightarrow A \rightarrow$	$D \rightarrow E$
$C \rightarrow$	$A \rightarrow B \rightarrow$	$D \rightarrow E$
$D \rightarrow$	$C \rightarrow E \rightarrow$	$A \rightarrow B$
$E \rightarrow$	$A \rightarrow C \rightarrow$	$B \rightarrow D$

This adjacency list becomes the basis of the intelligent partitioning. The entire region consists of paths starting with city  $A$  (chosen arbitrarily). If in each iteration the solution space is partitioned into  $M = 2$  subregions then the first subregion consists of all paths starting with either  $(A, E)$  or  $(A, C)$  as the first edge. The second subregion consists of all paths starting with  $(A, B)$  or  $(A, D)$  as the first edge.

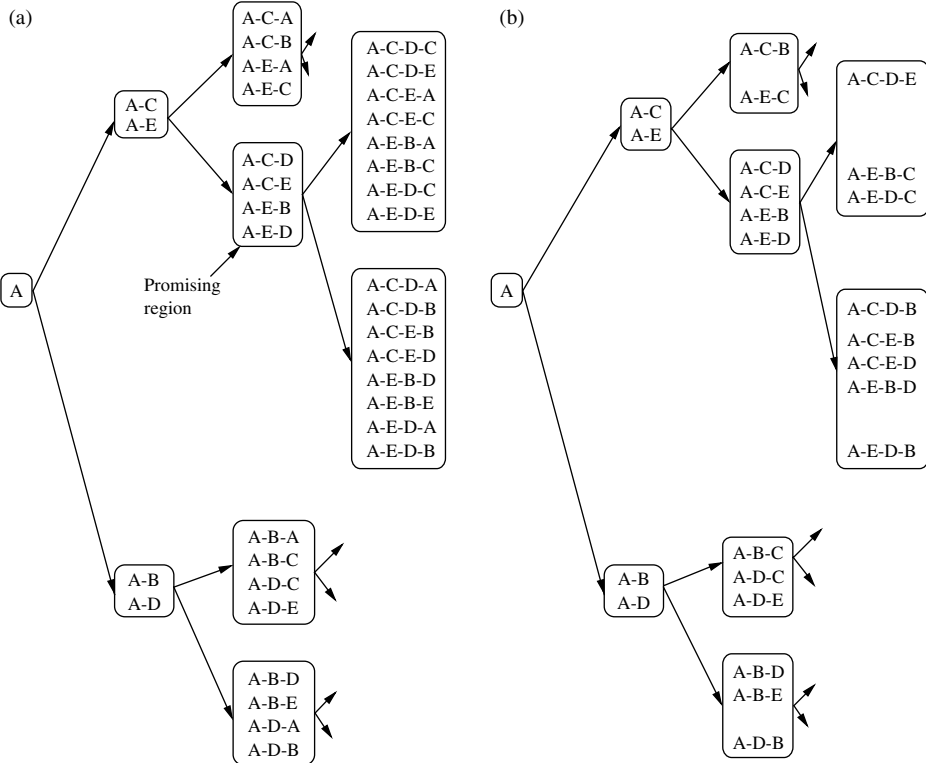
FIGURE 2. TSP example.



Now assume that the first subregion is chosen as the most-promising region. Then the first subregion of that region is the region that consists of all paths starting with  $(A, E, A)$ ,  $(A, E, C)$ ,  $(A, C, A)$ , or  $(A, C, B)$ . The second region can be read from the adjacency list in a similar manner. Note that one of these conditions creates an infeasible solution so there is no guarantee that all paths in a subregion will be feasible. It is, however, easy to check for feasibility during the sampling stage, and in fact this must always be done.

**5.1.3. Mathematical Programming for Intelligent Partitioning.** Sometimes solving a relaxation of the original problem may result in sufficient information to construct an intelligent partitioning. For example, we are solving a binary integer program (BIP),

FIGURE 3. Intelligent partitioning of the TSP.



defined by

$$\begin{aligned} z_{BIP} &= \min_x cx \\ Ax &\leq b \\ x &\in \{0, 1\}, \end{aligned} \tag{12}$$

that is,  $X = \{x \in \{0, 1\}: Ax \leq b\}$ . The linear program (LP) relaxation of BIP is

$$\begin{aligned} z_{LP} &= \min_x cx \\ Ax &\leq b \\ x &\in [0, 1]. \end{aligned}$$

This is an easy LP, which can be solved to obtain some optimal solution  $x_1^{LP}, x_2^{LP}, \dots, x_n^{LP}$ . In general  $x_i^{LP} \notin \{0, 1\}$  but the value can be taken as an indication of its importance. For example, if  $x_i = 0.95$  then it is intuitive that most of the good feasible solutions correspond to  $x_i = 1$  and most of the poor feasible solutions correspond to  $x_i = 0$ . On the other hand, if  $x_i = 0.5$  no such inference can be made. One possible intelligent partitioning for the BIP is therefore to order the variables according to the absolute deviation from one half, that is,

$$\left| \frac{1}{2} - x_{[1]} \right| \geq \left| \frac{1}{2} - x_{[2]} \right| \cdots \geq \left| \frac{1}{2} - x_{[n]} \right| \tag{13}$$

and start by partitioning  $\sigma(0) = X$  into two subregions

$$\begin{aligned} \sigma_1(0) &= \{x \in \{0, 1\}: x_{[1]} = 0, Ax \leq b\}, \\ \sigma_2(0) &= \{x \in \{0, 1\}: x_{[1]} = 1, Ax \leq b\}. \end{aligned}$$

We then continue to partition by fixing the remaining variables in the order (13) obtained by solving the LP relaxation. It is important to note that such intelligent partitioning is only a heuristic. It is possible that even though  $x_i$  is 0.95 in the relaxed solution, that  $x_i$  is 0 in the optimal solution. However, our empirical experience indicates that using such intuitive heuristics for intelligent partitioning is very effective in practice.

Similar to the LP relaxation for the BIP, solving any relaxation will reveal some information about what values are desirable for each variable. This can be utilized for developing an intelligent partitioning but the exact approach will in general depend on the specifics of the application.

**5.1.4. General Intelligent Partitioning.** Although intelligent partitioning methods will in general be application dependent, it may be possible to devise some general intelligent partitioning methods that perform well for a large class of problems. One such method would be based on generalizing the ideas from Ólafsson and Yang [4], where an intelligent partitioning is developed for the feature selection problem introduced in §4 above.

To develop a general intelligent-partitioning scheme, we use the idea of diversity from information theory, which is well-known in areas such as machine learning and data mining. For this purpose a solution need to be classified as being the same from the point of view of performance. A natural way to think about this is to say that two solutions are the same if there is little difference in their objective function values. Thus, a valid subregion with many solutions with significantly different objective function values is considered diverse, and vice versa. Diverse subregions are undesirable as it makes it difficult to determine which subregion should be selected in the next move.

To use traditional diversity measures, classify each solution into one category. First specify a small value  $\epsilon > 0$  such that two solutions  $x^1, x^2 \in X$  can be defined as having similar

performance if  $|f(x^1) - f(x^2)| < \epsilon$ . Then construct categories such that all solutions in each category are similar in this sense, and for any two categories there is at least one solution in each such that those two are dissimilar.

The following scheme can now be used to construct an intelligent partitioning:

1. Use random sampling to generate a set of  $M_0$  sample solutions.
2. Evaluate the performance  $f(x)$  of each one of these sample solutions, and record the average standard error  $\bar{s}^2$ .
3. Construct  $g(\bar{s}^2)$  intervals or categories for the sample solutions.
4. Let  $S_l$  be the frequency of the  $l$ th category in the sample set, and  $q_l = S_l/M_0$  be the relative frequency.
5. Let  $i = 1$ .
6. Fix  $x_i = x_{ij}$ ,  $j = 1, 2, \dots, m(x_i)$ .
7. Calculate the proportion  $p_l$  of solutions that falls within each category, and use this to calculate the corresponding entropy value:

$$E(i) = \sum_{l=1}^{g(\bar{s}^2)} q_l(i) \cdot I_l(i), \quad (14)$$

where

$$I_l(i) = - \sum_{j=1}^{g(\bar{s}^2)} p_{ij} \log_2(p_{ij}), \quad (15)$$

where  $p_{ij}$  is the proportion of samples with  $x_i = x_{ij}$ .

8. If  $i = n$ , stop; otherwise let  $i = i + 1$  and go back to step 6.

A high entropy value indicates high diversity, so it is desirable to partition by fixing the lowest entropy dimensions first. Thus, order the dimensions according to their entropy values

$$E(x_{[1]}) \leq E(x_{[2]}) \leq \dots \leq E(x_{[n]}), \quad (16)$$

and let this order determine the intelligent partition.

Note that we would apply this procedure before starting the actual NP method. This may be significant overhead but for difficult applications it is often worthwhile to expend such computational effort developing intelligent partitioning. This imposes a useful structure that can hence improve the efficiency of the NP method itself, which often vastly outweighs the initial computational overhead.

## 5.2. Randomly Generating Feasible Solutions

In addition to using domain understanding to devise partitioning that imposes a structure on the feasible region, the other major factor in determining the efficiency of the NP method is how feasible solutions are generated from each region. The *pure NP algorithm* prescribes that this should be done randomly but there is a great deal of flexibility in both how those random samples should be generated and how many random samples should be obtained.

The goal should be for the algorithm to frequently make the correct move, that is, either move to a subregion containing a global optimum or backtrack if the current most-promising region does not contain a global optimum. In the theoretical ideal, the correct move will always be made if the best feasible solution is generated in each region. This is of course not possible except for trivial problems, but in practice the chance of making the correct move can be enhanced by (i) biasing the sampling distribution so that good solutions are more likely to be selected, (ii) incorporating heuristic methods to seek out good solutions, and (iii) obtaining a sufficiently large sample. We will now explore each of these issues.

**5.2.1. Biased Random Sampling.** We illustrate some simple random sampling methods for generating feasible solutions to the TSP. Assume that the generic partitioning (Figure 1) is used and the current most-promising region is of depth  $k$ . This means that the first  $k$  edges in the TSP tour have been fixed. Generating a sample solution from this region entails determining the  $n - k$  remaining edges. One approach would be simply to select the edges consecutively, such that each feasible edge has equal probability of being selected (*uniform sampling*). However, this approach may not give good results in practice for the same reason why a generic partitioning may be inefficient; that is, uniform sampling considers only the solution space itself.

To incorporate the objective function into the sampling, consider the following biased sampling schemes. At each iteration, weights  $(w_{j_{i-1}, j_l})$  are calculated and assigned to each of the remaining cities that need to be determined. The weight is inversely proportional to the cost of the edge from city  $j_l$  to the city  $j_{i-1}$ . Specifically we can select the weights as

$$w_{j_{i-1}, j_l} = \frac{(c_{j_{i-1}, j_l})^{-1}}{\sum_{h=i}^n (c_{j_{i-1}, j_h})^{-1}},$$

where the weights have been normalized so that the sum adds to one. The next edge can now be selected by uniformly generating a number  $u$  between zero and one and comparing this number with the weights, that is, if

$$\sum_{m=i}^{i^*} w_{j_{i-1}, j_m} \leq u < \sum_{m=i}^{i^*+1} w_{j_{i-1}, j_m}$$

then city  $i^*$  is selected and  $(j_{i-1}, j_{i^*})$  becomes the next edge.

This is a randomized procedure for generating feasible sample solutions. Each edge has a positive probability of being selected next and hence for every region all of the feasible tours in the region have a positive probability of being generated using this procedure. However, this probability is no longer uniform. The probability has been biased so that low-cost edges are selected with higher probability and tours with many low-cost edges are therefore generated with higher probability. In our computational experience such biased sampling approaches will tend to significantly improve the efficiency of the NP method.

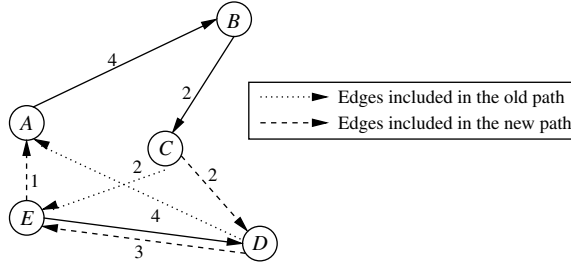
It is important to note that sampling is very flexible when dealing with very hard constraints. Say for example that the TSP has some additional constraints such as time windows or restrictions on the order in which the cities must be visited. Such constraints can be very difficult to deal with using mathematical programming techniques, but only a minor modification of the sampling procedure would be needed to assure that only feasible solutions are generated. Thus, the use of sampling makes the NP method extremely effective when dealing with complex constraints. Many problems that arise in complex applications have both easy and complex constraints; with the NP method it is possible to use sampling to deal with the complex constraints and to use exact methods to deal with the easy constraints.

**5.2.2. Incorporating Heuristics in Generating Solutions.** In addition to biasing the sampling distribution, it may be possible to quickly generate good feasible solutions by applying (randomized) heuristic search. This can for example be done with a simple local search such as in the following algorithm that generates  $N$  feasible solutions:

1. Obtain one random sample using either uniform or weighted sampling.
2. Obtain  $N_j - 1$  more samples by making small perturbations of the sample generated in the first step.

To illustrate this approach consider how the local search sampling can be applied to the TSP. The second step could for example involve randomly selecting two edges and connecting the first vertex of the first edge to the first vertex of the second edge, and connecting the second

FIGURE 4. The second step in the two-step sampling scheme.



vertex of the first edge to the second vertex of the second edge. This technique is similar to a two-opt exchange but does not consider if the performance is improved by this exchange. Other, more complicated variants, with more than two edges selected at random, are easily obtained in a similar fashion. Step 2 is further illustrated in Figure 4. A sample obtained in step 1 is shown with the combination of solid and dotted lines. Then in step 2, select the edges  $(C, E)$  and  $(D, A)$  at random and replace them with the edges  $(C, D)$  and  $(E, A)$ . The new edges are shown as dashed lines in Figure 4. Clearly this procedure provides us with a new sample solution with relatively little effort.

Instead of using an application specific heuristic such as the one above, any other heuristic could be incorporated in the same manner. For example, we can incorporate the genetic algorithm (GA) cross-over operator into the sampling as follows for the TSP problem above. Start by randomly generating two sample solutions, e.g.,  $(A, E, C, D, B)$  and  $(E, A, C, B, D)$  called the parents. Select a cross-over point, say after the second city in the tour, and generate two new solutions called the children, namely  $(A, E, C, B, D)$ , where the first two elements come from the first parent and the other three from the second, and  $(E, A, C, D, B)$ , where the first two elements come from the second parent and the other three from the first. Of course the cross-over operator can be applied more than once and the other main GA operator of mutation can similarly be incorporated into the NP framework. Thus, an entire GA search can easily be used to generate high-quality feasible solutions from each region and the same is true for any other heuristic that is thought to perform well for a particular application.

**5.2.3. Incorporating Mathematical Programming.** At first glance it may seem that mathematical-programming methods would not be very useful for generating feasible solutions within the NP method. The focus of such methods for discrete problems is the generation of lower bounds that rarely correspond to feasible solutions. However, it turns out that mathematical-programming methods can indeed be very useful for generating feasible solutions and incorporating them for this purpose can significantly improve the efficiency of the NP method.

There are two distinct ways in which mathematical programming can be used to generate feasible solutions. First, the solution to a relaxation of the original Problem (2) can be used to bias the sampling. The basic idea is for solutions that are similar to the optimal solutions for the relaxed problem to be sampled with higher probability. To illustrate, we consider again the generic BIP discussed above, namely,

$$\begin{aligned} z_{BIP} = \min_x & \quad cx \\ & \quad Ax \leq b \\ & \quad x \in \{0, 1\}. \end{aligned}$$

The LP relaxation can be solved to obtain some optimal solution  $x_1^{LP}, x_2^{LP}, \dots, x_n^{LP}$  and we can then bias the sampling distribution according to these values. For example, for any  $x_i$  we can take the sampling distribution to be

$$P[x_i = 1] = x_i^{LP}, \quad (17)$$

$$P[x_i = 0] = 1 - x_i^{LP}. \quad (18)$$

Thus, if a particular variable  $x_i$  is close to one in the LP relaxation solution then it is one with high probability in the sample solution, and vice versa.

The second approach to incorporating mathematical programming into the generation of feasible solutions applies when the Problem (2) can be decomposed into two parts, one that is easy from a mathematical-programming perspective and one that is hard. For such a problem it is impossible in practice to use mathematical programming to solve the entire problem, but when solving the problem using the NP method we can take advantage of the fact that mathematical programming can effectively solve a partial problem. Specifically, we can use sampling to generate partial solutions that fix the hard part of the problem and then complete the solution by solving a mathematical program. Because the mathematical-programming output is optimal given the partial sample solution, this process can be expected to result in higher quality feasible solutions than if the entire solution was obtained using sampling. On the other hand, the process of generating a sample solution is still fast because the difficult part of the problem is handled using sampling. This first part can incorporate any biased sampling approach or heuristics, and the combined procedure for generating feasible solutions is therefore a prime example of how mathematical programming and heuristics search complement each other when both are incorporated into the NP framework.

**5.2.4. Determining the Total Sampling Effort.** As might be expected, incorporating special structure and heuristics to generate feasible solutions results in finding better solutions. This in turn leads to the NP algorithm selecting the correct move more frequently and hence improves the efficiency of the search. The question still remains of how many sample solutions are needed to make the correct choice with a sufficiently large probability.

It is possible to connect the minimum required probability of making a correct move to how likely we want it to be that the optimal solution is eventually found (Ólafsson [3]). The number of feasible sample solutions required to assure this minimum probability depends on the variance of the performance of the generated solutions. In the extreme case, if the procedure that is used to generate feasible solutions always results in a solution that has the same performance then there is no advantage to generating more than one solution. Vice versa, if the procedure leads to solutions that have greatly variable performance then it may be necessary to generate many solutions to obtain a sufficiently good estimate of the overall performance of the region.

This observation motivates the following two-stage sampling approach. In the first stage generate a small number of feasible solutions using uniform sampling, weighted sampling, local search sampling, or any other appropriate method for generating sample solutions. Calculate the variance of the performance of these solutions and then apply statistical selection techniques to determine how many total samples are needed to achieve the desired results.

### 5.3. Backtracking and Initialization

Another critical aspect of the NP method is the global perspective it achieves by generating solutions from the complimentary region and backtracking if necessary. Specifically, if the best feasible solution is found in the complimentary region, this is an indication that the

incorrect move was made when  $\sigma(k)$  was selected as the most-promising region so the NP algorithm backtracks by setting  $\sigma(k+1) = \sigma(k-1)$ .

Backtracking is usually very easy to implement because some type of truncating is usually sufficient and we do not need to keep track of the last most-promising region. For example, in a five-city TSP problem the current most-promising region is defined by the sequence of cities  $B \leftarrow D \leftarrow C$  with the remaining cities undecided. Thus, the current most-promising region can be written as

$$\sigma(k) = \{(B, D, C, x_4, x_5): x_4, x_5 \in \{A, E\}, x_4 \neq x_5\}.$$

If backtracking is indicated, then the next most-promising region becomes

$$\sigma(k+1) = \{(B, D, x_3, x_4, x_5): x_i \in \{A, C, E\}, x_i \neq x_j \text{ if } i \neq j\}.$$

Thus, backtracking is simply achieved by truncating the sequence that defines the current most-promising region. Similar methods can be used for most other problems making backtracking possible with very little or no overhead.

Unless otherwise noted we will always assume that backtracking is done as above, that is,  $\sigma(k+1) = \sigma(k-1)$ . However, it is clear that it is possible to backtrack in larger steps. For example by truncating two or three cities from the sequence that defines the current most-promising region in the TSP application. The advantage is that this would enable the algorithm to reverse a sequence of incorrect move more easily. On the other hand it reverses several NP moves based on the results from one iteration, and if the backtracking turns out to be incorrect, it would take several moves to get back to the previous point. For this reason we do not advocate this approach—rather the focus should be on making each move correctly with high probability. As noted above, this can be done by developing intelligent partitioning and using biased sampling and heuristics to generate high-quality feasible solutions. In other words, by incorporating domain knowledge and special structure, incorrect moves become infrequent and it is hence always sufficient to simply backtrack one step to the previous most-promising region.

As we will see in §6, backtracking assures that the NP method converges to the globally optimal solution and does not become stuck at a local optimum. However, in practice excessive backtracking indicates an inefficient implementation of the NP method. If backtracking is correctly called for, this implies that at least one incorrect move was previously made. Thus, by monitoring the amount of backtracking it is possible to design adaptive NP algorithms. If excessive backtracking is observed this indicates that more effort is needed to evaluate regions before a choice is made. More or higher quality feasible solutions should hence be generated before a choice is made, which can be done by uniform random sampling, local search sampling, or any other appropriate method.

Finally, we note that although we will generally assume that the initial state of the search is to let the entire feasible region be the most promising, that is,  $\sigma(0) = X$ , this does not always need to be the case. For example, if time is limited and it is important to generate good solutions quickly (and then possibly continue to generate better solutions), it may be worthwhile to initialize the search and set  $\sigma(0) = \eta$ , where  $\eta \in \Sigma \setminus \{X\}$  is a partial solution determined using a heuristic or domain knowledge.

#### 5.4. Promising Index

The final aspect of the NP method is the promising index that is used to select the next most-promising region. This promising index should be based on the sample information obtained by generating feasible sample solutions from each region but other information could also be incorporated.



Unless otherwise noted we will assume that the promising index for a valid region  $\sigma \in \Sigma$  is based only on the set of feasible solutions  $D_\sigma$  that are generated from this region, and it is taken to be

$$I(\sigma) = \min_{x \in D_\sigma} f(x). \quad (19)$$

However, other promising indices may be useful. For example, it is easy to solve a relaxation of the problem (1) using mathematical-programming methods and hence obtain a lower bound  $\underline{f}(\sigma)$  on the objective function. This lower bound can be combined with the upper bound  $\min_{x \in D_\sigma} f(x)$  into a single promising index

$$I(\sigma) = \alpha_1 \cdot \underline{f}(\sigma) + \alpha_2 \cdot \min_{x \in D_\sigma} f(x), \quad (20)$$

where  $\alpha_1, \alpha_2 \in \mathbf{R}$  are the weights given to the lower bound and upper bound, respectively. The lower bound could be obtained using any standard mathematical-programming method that is appropriate, such as linear programming (LP) relaxation, Lagrangian relaxation, or a application specific COP relaxation, such as relaxing the subset elimination constraints for a TSP.

However, for large-scale complex discrete problems where the NP method is the most useful, it is often not possible to obtain useful lower bounds. In such cases, a probabilistic bound may be useful. A probabilistic bound may be obtained as follows. The process by which feasible solutions are generated from each region attempts to estimate the extreme point of the region. This may or may not include a heuristic search. Let  $x^*(\sigma)$  denote the true extreme point (minimum) of a valid region  $\sigma \in \Sigma$ . Then the extreme performance  $\hat{f}^*(\sigma) = f(x^*(\sigma))$  is estimated as

$$\hat{f}^*(\sigma) = \min_{x \in D_\sigma} f(x), \quad (21)$$

where the set of feasible solutions  $D_\sigma$  could be based on pure random sampling, applying a local search heuristic to an initial random sample, or applying a population-based heuristic such as a genetic algorithm to a random initial population. Now assume that we generate several such sets  $D_\sigma^1, \dots, D_\sigma^n$  and calculate the corresponding extreme value estimates  $\hat{f}_1^*(\sigma), \dots, \hat{f}_n^*(\sigma)$ . It is then possible to construct an overall estimate

$$\hat{f}_{\min}^* = \min_i \min_{x \in D_\sigma^i} f(x) \quad (22)$$

and a  $1 - \alpha$  confidence interval  $[l(D_\sigma^1, \dots, D_\sigma^n), u(D_\sigma^1, \dots, D_\sigma^n)]$  for the extreme value, that is,

$$P[f^*(\sigma) \in [l(D_\sigma^1, \dots, D_\sigma^n), u(D_\sigma^1, \dots, D_\sigma^n)]] = 1 - \alpha.$$

The left end of the confidence interval may hence be viewed as a *probabilistic lower bound* for the performance (extreme point) of the region

$$\underline{\hat{f}}(\sigma) = l(D_\sigma^1, \dots, D_\sigma^n). \quad (23)$$

This can then be incorporated into the promising index similar to the exact bound above, namely,

$$I(\sigma) = \alpha_1 \cdot \underline{\hat{f}}(\sigma) + \alpha_2 \cdot \hat{f}_{\min}^*. \quad (24)$$

The estimation of the confidence interval is typically based on the assumption that the extreme values follow a Weibul distribution.

In addition to the exact or probabilistic lower bounds, many other considerations would be incorporated into the promising index. For example cost penalties obtained from other regions or the variability of performance within the region could be added. The latter would be of particular interest if we want to obtain not only good solutions but also robust solutions; that is, small changes in this type of solution will not greatly change the performance. In general, any domain knowledge or application-appropriate technique can in a similar manner be incorporated into the promising index and used to guide the search more efficiently.

## 6. Convergence Properties

In this section we analyze the convergence of the NP algorithm. The first main convergence result is that the NP algorithm converges to an optimal solution of any COP in finite time. The proof is based on a Markov chain analysis that utilizes the fact that the sequence of most-promising regions is an absorbing Markov chain and the set of optimal solutions corresponds exactly to the absorbing states. Because the state space of the Markov chain is finite it is absorbed in finite time. With some additional assumptions, this result can be generalized to problems with infinite countable feasible regions (such as IPs) and even continuous feasible regions (such as MIPs).

The second main result is that the time until convergence can be bounded in terms of the size of the problem and the probability of making the correct move. This result will show that the expected number of iterations grows slowly in terms of the problem size, which explains why the NP algorithm is effective for large-scale optimization. On the other hand, as the probability of making the correct move decreases, the expected number of iterations increases exponentially. This underscores the need to increase this probability by incorporating special structure in both the partitioning and the method used for generating feasible solutions.

The main results are presented in this section without proofs. For more details and complete proofs we refer the reader to Shi and Ólafsson [7].

### 6.1. Finite Time Convergence for COPs

In this section we assume that the NP algorithm is applied to a combinatorial optimization problem (COP). We start by formally stating the Markov property.

**Proposition 1.** *Assume that the partitioning of the feasible region is fixed and  $\Sigma$  is the set of all valid regions. The stochastic process  $\{\sigma(k)\}_{k=1}^{\infty}$ , defined by the most-promising region in each iteration of the pure NP algorithm, is a homogeneous Markov chain with  $\Sigma$  as state space.*

The next result needed is that the optimal solution(s) are absorbing states.

**Proposition 2.** *Assume that the partitioning of the feasible region is fixed and  $\Sigma$  is the set of all valid regions. A state  $\eta \in \Sigma$  is an absorbing state for the Markov chain  $\{\sigma(k)\}_{k=1}^{\infty}$  if and only if  $d(\eta) = d^*$  and  $\eta = \{x^*\}$ , where  $x^*$  is a global minimizer of the original problem, as it solves Equation (1) above.*

The following theorem is now immediate.

**Theorem 1.** *The NP algorithm converges almost surely to a global minimum of the optimization problem given by Equation (1) above. In mathematical notation, the following equation holds.*

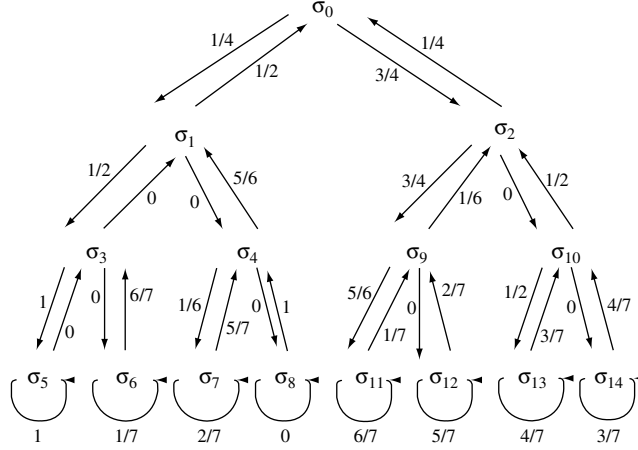
$$\lim_{k \rightarrow \infty} \sigma(k) = \{x^*\} \quad \text{a.s.}, \quad (25)$$

where

$$x^* \in \arg \min_{x \in \Theta} f(\theta).$$

It is evident that the transition probabilities of the Markov chain depend on the partitioning and how feasible solutions are generated. This in turn will determine how fast the Markov chain converges to the absorbing state and hence it determines the performance of the NP algorithm. It is therefore clearly of great practical importance to develop intelligent partitioning for specific problem structures of interests. The following examples illustrate how to calculate the transition probabilities and how they are dependent on the partitioning.

FIGURE 5. Transition probabilities for the Markov chain.



**Example 2.** Consider an example with  $\sigma_0 = \{1, 2, 3, 4, 5, 6, 7, 8\}$  and 14 other valid regions are defined by partitioning each region into two subregions ( $M_{\max} = 2$ ). This partitioning can be represented by the Markov chain in Figure 5, which also shows the transition probabilities given the function values described below.

Further assume that in each iteration, we uniformly generate one point in each region at random ( $N_{\max} = 1$ ), and that the relative ranking of the function values is give as follows,

$$f_1 < f_5 < f_6 < f_7 < f_8 < f_3 < f_2 < f_4.$$

To simplify the notation we define  $f_i = f(\sigma_i)$ ,  $i = 1, 2, \dots, 8$ . Now the transition probabilities can be easily calculated. For example, if the current region is  $\sigma_0$ , then look at regions  $\sigma_1 = \{1, 2, 3, 4\}$  and  $\sigma_2 = \{5, 6, 7, 8\}$ ; the only way  $\sigma_1$  can be found to be better is if point 1 is picked up at random. Because it is assumed that points are selected uniformly, the probability of this is  $1/4$ . Therefore we get  $P_{\sigma_0 \sigma_1} = 1/4$  and it follows that because  $P_{\sigma_0 \sigma_2} = 1 - P_{\sigma_0 \sigma_1}$ , one gets  $P_{\sigma_0 \sigma_2} = 3/4$ .

Now assume that the current region is  $\sigma_1$  and look at the regions  $\sigma_3 = \{1, 2\}$ ,  $\sigma_4 = \{3, 4\}$ , and  $X \setminus \sigma_1 = \{5, 6, 7, 8\}$ . It is clear that no matter which point is selected from  $\sigma_4$ , one will always generate a better point from  $X \setminus \sigma_1$ . We can therefore conclude that  $P_{\sigma_1 \sigma_4} = 0$ .

Now  $\sigma_3$  can only be considered best if solution 1 is randomly selected from that region. Hence  $P_{\sigma_1 \sigma_3} = \frac{1}{2}$ . Because there are only three regions that have a positive transition probability, the third is now also determined as  $P_{\sigma_1 \sigma_0} = \frac{1}{2}$ .

In a similar manner we can for example get  $P_{\sigma_3 \sigma_5} = 1$  and  $P_{\sigma_5 \sigma_5} = 1$ . This shows  $\sigma_5$  is an absorbing state, which is in agreement with being the global optimum. The remaining transition probabilities are calculated the same way, and they are shown in Figure 5. It is immediate from the figure that there are no other absorbing states.

In the next example the transition probabilities are calculated for the same problem, but with a different partitioning.

**Example 3.** Now assume the same problem as in the preceding example, except that in each iteration we partition into four subregions ( $M_{\max} = 4$ ). Then get the following regions:  $\sigma_0 = \{1, 2, 3, 4, 5, 6, 7, 8\}$ ,  $\sigma_1 = \{1, 2\}$ ,  $\sigma_2 = \{3, 4\}$ ,  $\sigma_3 = \{5, 6\}$ ,  $\sigma_4 = \{7, 8\}$ ,  $\sigma_5 = \{1\}$ ,  $\sigma_6 = \{2\}$ , and so forth. It is immediately clear that  $P_{\sigma_0, \sigma_1} = \frac{1}{2}$  and  $P_{\sigma_1, \sigma_5} = 1$ .

It can be concluded that there is 50% probability of getting to the global optimum in only two iterations. This is clearly much better than with the previous partition, and illustrates that the partitioning of the feasible region affects how fast the algorithm converges. It should be noted however, that in this second example, four to five function evaluations are done in each step that is not at maximum depth, but in the first example, only two to three function evaluations were done in each step.

The last two examples clearly illustrate how the partitioning influences the transition probabilities and hence the speed of convergence. It is also clear that the transition probabilities depend on the number of points sampled from each region in each iteration.

**Example 4.** Alternatively, it is still possible to partition into two subregions for each region, but to do it in a different manner. For example, we let  $\eta_1 = \{1, 5, 6, 7\}$ ,  $\eta_2 = \{2, 3, 4, 8\}$ ,  $\eta_3 = \{1, 5\}$ ,  $\eta_4 = \{6, 7\}$ ,  $\eta_5 = \{1\}$ , and so forth. Then  $P_{\eta_0, \eta_1} = 1$ ,  $P_{\eta_1, \eta_3} = 1$ , and  $P_{\eta_3, \eta_5} = 1$ . Hence the NP method converges to the global optimum in three iterations with probability one. Such *optimal partitioning* always exists but would normally require too much computational effort to obtain. On the other hand, it shows how the NP method is heavily dependent on the partitioning and provides motivation for finding intelligent partitioning strategies.

## 6.2. Solving IPs and MIPs

It should be noted that the analysis in the previous section is identical for IP problems and other problems that may have countable infinite feasible region, except that the Markov chain now has an infinite state space. For MIPs where the feasible region is uncountable but bounded, similar analysis can also be done but some additional conditions are required.

It is not difficult to verify that the NP method generates a Markov chain when applied to the optimization problem with uncountable but bounded feasible region. It is also easy to see that the only states containing a global optimum can be absorbing states. However, some additional conditions are needed to prove that all such states will be absorbing.

The basic condition for a state containing a global optimum to be absorbing is that it is sufficiently small and/or the method for generating feasible solutions is sufficiently powerful. In particular, if the method for generating solutions will always find the global optimum when applied to the smallest state containing this optimum, then this state is absorbing. If this is true for at least one global optimum, the NP method will converge in finite time when solving MIPs.

## 6.3. Time Until Convergence

We now return to analyzing the NP method for solving COPs, which are the main problem domain studied in this tutorial. It has been established that the Markov chain of most-promising regions will eventually get absorbed at a global optimum. Although such convergence results are important it is of much interest to establish bounds on how many iterations, and consequently how many function evaluations, are required before the global optimum is found. In this section such bounds are established for the expected number of iterations until absorption. It is assumed that  $\Sigma$  is finite, which is the case for all COPs and for MIPs if the partitioning is defined appropriately. To simplify the analysis it is assumed that the global optimum is unique.

Let  $Y$  denote the number of iterations until the Markov chain is absorbed and let  $Y_\eta$  denote the number of iterations spent in state  $\eta \in \Sigma$ . Because the global optimum is unique, the Markov chain first spends a certain number of iterations in the transient states and when it first hits the unique absorbing state, it never visits any other states. Hence it is sufficient to find the expected number of visits to each of the transient states. Define  $T_\eta$  to be the hitting time of state  $\eta \in \Sigma$ , i.e., the first time that the Markov chain visits the state. Also let  $E$  denote an arbitrary event and let  $\eta \in \Sigma$  be a valid region. We let  $P_\eta[E]$  denote the probability of event  $E$  given that the chain starts in state  $\eta \in \Sigma$ .

It will be convenient to consider the state space  $\Sigma$  as consisting of three disjoint subsets as follows. Let  $\sigma^*$  be the region corresponding to the unique global optimum. Define  $\Sigma_1 = \{\eta \in \Sigma \setminus \{\sigma^*\} \mid \sigma^* \subseteq \eta\}$  and  $\Sigma_2 = \{\eta \in \Sigma \mid \sigma^* \not\subseteq \eta\}$ . Then  $\Sigma = \{\sigma^*\} \cup \Sigma_1 \cup \Sigma_2$  and these three sets are disjoint. Using this notation, we can now state the following result, which relates the expected time to the hitting probabilities.

**Theorem 2.** *The expected number of iterations until the NP Markov chain gets absorbed is given by*

$$E[Y] = 1 + \sum_{\eta \in \Sigma_1} \frac{1}{P_\eta[T_{\sigma^*} < T_\eta]} + \sum_{\eta \in \Sigma_2} \frac{P_X[T_\eta < \min\{T_X, T_{\sigma^*}\}]}{P_\eta[T_X < T_\eta] \cdot P_X[T_{\sigma^*} < \min\{T_X, T_\eta\}]} \quad (26)$$

The expected number of iterations (26) depends on both the partitioning and how feasible sample solutions are generated, as well as the structure of the problem itself. Calculating this expectation exactly is therefore complicated. However, with some additional assumptions it is possible to find useful bounds on the expected number of iterations.

Assume that  $P^*$  is a lower bound on the probability of selecting the correct region. We refer to the probability  $P^*$  as the minimum success probability. The next theorem provides an upper bound for the expected time until the NP algorithm converges in terms of  $P^*$  and the size of the feasible region as measured by  $d^*$ .

**Theorem 3.** *Assume that  $P^* > 0.5$ . The expected number of iterations until the NP Markov chain gets absorbed is bounded by*

$$E[Y] \leq \frac{d^*}{2P^* - 1}. \quad (27)$$

Note that (27) grows only linearly in  $d^*$ . Furthermore, as for any other tree, the depth  $d^*$  of the partitioning tree is a logarithm function of the input parameter(s). In other words, assume for simplicity reasons that there is a single input parameter  $n$ , then (27) provides a bound on the expected number of iterations that is  $O(\log n)$ . This shows that the expected number of iterations required by the NP algorithm grows very slowly in the size of the problem and this partially explains why the NP algorithm is very effective for solving large-scale optimization problems.

On the other hand, the bounds (27) grow exponentially as  $P^* \rightarrow \frac{1}{2}$ . This is further illustrated in Table 1 that shows the bounds for several problem sizes ( $d^*$ ) and the values for the minimum success probability ( $P^*$ ). Clearly, the number of expected iterations increases rapidly as the success probability decreases.

These results underscore the previously made statement that the efficiency of the NP algorithm depends on making the correct move frequently. This success probability depends in turn on both the partitioning and the method for generating feasible solutions. For any practical application it is therefore important to increase the success probability by developing intelligent partitioning methods, incorporating special structure into weighted sampling, and applying randomized heuristics to generate high-quality feasible solutions.

TABLE 1. Bounds on the expected time until convergence.

Success prob. (%)	Maximum depth ( $d^*$ )				
	2	5	10	20	30
55	20	50	100	200	300
60	10	25	50	100	150
65	7	17	33	67	100
70	5	13	25	50	75
75	4	10	20	40	60
80	3	8	17	33	50
85	3	7	14	29	43
90	3	6	13	25	38
95	2	6	11	22	33

## 7. Conclusions

In this tutorial we have discussed the NP method, a relatively new powerful metaheuristic for solving large-scale discrete optimization problems. The method systematically partitions the feasible region into subregions and moves from one region to another based on information obtained by randomly generating feasible sample solutions from each of the current regions. The method keeps track of which part of the feasible region is the most promising in each iteration and the number of feasible solutions generated; hence the computational effort is always concentrated in this most-promising region.

The efficiency of the NP algorithm depends on frequently making the correct move. This success probability depends in turn on both the partitioning and the method for generating feasible solutions. For any practical application it is therefore important to increase the success probability by developing intelligent partitioning methods, incorporating special structure into weighted sampling, and applying randomized heuristics to generate high-quality feasible solutions.

The NP method has certain connections to standard mathematical-programming techniques such as branch-and-bound. However, the NP method is primarily useful for problems that are either too large or too complex for mathematical programming to be effective. But even for such problems mathematical-programming methods can often be used to solve either a relaxed problem or a subproblem of the original and these solutions can be effectively incorporated into the NP framework.

## References

- [1] W. D. D'Souza, R. R. Meyer, and L. Shi. Selection of beam orientations in intensity-modulated radiation therapy using single-beam indices and integer programming. *Physics in Medicine and Biology* 49:3465–3481, 2004.
- [2] M. Gendreau and J. Potvin. Metaheuristics in combinatorial optimization. *Annals of Operations Research* 140:189–213, 2005.
- [3] S. Ólafsson. Two-stage nested partitions method for stochastic optimization. *Methodology and Computing in Applied Probability* 6:5–27, 2004.
- [4] S. Ólafsson and J. Yang. Intelligent partitioning for feature selection. *INFORMS Journal on Computing* 17(3):339–355, 2005.
- [5] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal* 27:379–423, 1948.
- [6] L. Shi and S. Ólafsson. Nested partitions method for global optimization. *Operations Research* 48:390–407, 2000.
- [7] L. Shi and S. Ólafsson. *Nested Partitions Optimization: Methodology and Applications*. *International Series in Operations Research & Management Science*, Vol. 109. Springer, New York, 2007.
- [8] L. Shi, S. Olafasson, and Q. Qun. A new hybrid optimization algorithm. *Computer and Industrial Engineering* 36:409–426, 1999.
- [9] L. Wolsey. *Integer Programming*. John Wiley & Sons, New York, 1998.

# Computational Global Optimization

*Leon S. Lasdon*

Information, Risk, and Operations Management Department, McCombs Business School,  
University of Texas at Austin, Austin, Texas 78712, lasdon@mail.utexas.edu

*János D. Pintér*

Pintér Consulting Services Inc., Halifax, Nova Scotia, Canada B3M 1J2, jdpinter@hfx.eastlink.ca

**Abstract** The objective of global optimization (GO) is to find the best possible solution in nonlinear models that possess multiple—local and global—optima. GO has become a subject of growing interest in recent decades. The theoretical results regarding practically important GO model types and generic algorithmic solution approaches have been followed by software implementations that are now used to handle a large variety of applications. In this article, we introduce the continuous GO model, with reference to some of its special cases. Next, we classify the most prominent exact and heuristic solution approaches, and then review software implementations in various modeling environments. We also highlight a range of GO applications, and point toward present and future challenges.

**Keywords** nonlinear systems analysis and management; global optimization models and solution strategies; modeling environments and global solver implementations; numerical examples; scientific and engineering applications

---

## 1. Introduction

Nonlinearity plays a fundamental role in the development of natural and man-made objects, formations, and processes. Consequently, nonlinear descriptive models are relevant across a vast range of scientific and engineering studies. For related discussions that illustrate this point (in the context of various scientific and engineering disciplines), consult for instance Aris [1], Casti [9], Fritzson [19], Gao et al. [21], Gershenfeld [22], Hansen and Jørgensen [26], Jacob [31], Lopez [40], Mandelbrot [41], Murray [52], Papalambros and Wilde [58], Rich [74], Schittkowski [77], Schroeder [78], Stewart [79], Stojanovic [80], and Wolfram [90].

The numerical solution of a process control or systems optimization model that incorporates an underlying nonlinear system description requires nonlinear optimization (NLO) techniques. Traditionally, such methods have been of limited scope, because NLO has mostly been applied to convex models. In general, from a given starting point (supplied by an expert or chosen in some way by the modeler) a local search method typically leads to the corresponding locally best solution. Under specific structural assumptions—such as convexity and some of its relatively mild generalizations—this is the true solution of the problem. Local scope NLO methods have been widely discussed and applied (using computers) for over fifty years. Among the numerous topical textbooks that discuss traditional NLO we mention here Bazaraa et al. [2], Bertsekas [4], Boyd and Vandenberghe [7], Chong and Zak [10], Hillier and Lieberman [27], Lasdon [36], and Nocedal and Wright [55].

Local scope search has its inherent limitations when the model to solve has multiple optima. In such cases, one would like to find the global optimum which represents the “absolutely best” solution, as opposed to finding a “locally best” solution. For illustration, let us mention a few general model types that are frequently met in practice, and require global scope search.

The first broad category consists of models that provably do not satisfy the standard structural conditions (such as the convexity of the feasible region and the strict convexity of the objective function). For example, models that include nonlinear equality constraints are nonconvex. Such models arise in studies of material balances, nonlinear physical property relations, blending (mixing) equations, nonlinear processes, and economic equilibria, to name a few. Another frequent source of nonconvexity is a nonconvex—typically concave—cost or utility function. Such functions are often used to describe production costs, with consideration to economies of scale. As a third example, models that include trigonometric functions (that typically represent physical or other periodicities) are, as a rule, nonconvex and hence multimodal. Let us also point out that models that contain discrete-valued variables are also nonconvex. Such models are ubiquitous e.g., in a large variety of vehicle routing, staff scheduling, and job-shop sequencing problems.

The second important category consists of models in which it is difficult or impossible to verify whether they are convex or not. Let us recall here that the standard definition of convexity is not constructive, because it would require us to verify the convexity inequality of a given objective function, for all possible pairs of feasible points. Therefore with the exception of certain function classes (including some—but far from all—important elementary functions, as well as the family of positive semi-definite quadratic functions), the possible convexity of nonlinear functions often remains unknown. As a result, the convexity of practically important model instances—specifically including many dynamic and stochastic models, as well as models that contain embedded numerical procedures (systems of differential equations, integration, simulation, and so on)—cannot be simply postulated or verified. A practically most relevant example is general nonlinear regression (the selection of model parameters by fitting a given descriptive model type to data); the typical (least overall error or maximum likelihood) objectives often have an unknown number of local minima.

The objective of global optimization (GO) is to find the “absolutely best” solution of nonlinear optimization models, when they are provably nonconvex or suspect to multimodality. In the present discussion, we shall consider the general continuous global optimization (CGO) model defined by the following ingredients:

- $x$  decision vector, an element of the real Euclidean  $n$ -space  $\mathbf{R}^n$ ;
- $l, u$  explicit, finite  $n$ -vector bounds of  $x$  that define a “box” in  $\mathbf{R}^n$ ;
- $f(x)$  continuous objective function,  $f: \mathbf{R}^n \rightarrow \mathbf{R}$ ;
- $g(x)$   $m$ -vector of continuous constraint functions,  $g: \mathbf{R}^n \rightarrow \mathbf{R}^m$ .

Applying this notation, the CGO model is stated as

$$\min f(x) \tag{1}$$

$$x \in D := \{x: l \leq x \leq u \mid g(x) \leq 0\}. \tag{2}$$

In (2) all vector inequalities are interpreted component-wise:  $l, x, u$ , are all  $n$ -component vectors and the zero denotes an  $m$ -component vector. The set of the additional (general) constraints  $g$  could be empty, thereby leading to a box-constrained GO model. Formally more general optimization models that include also the  $=$  and  $\geq$  constraint relations, and/or explicit lower bounds on the constraint function values can simply be reduced to the model form (1)–(2).

The CGO model is very general; in fact, it evidently subsumes linear programming and convex nonlinear programming models, under corresponding additional specifications. Furthermore, CGO also formally subsumes the entire class of pure and mixed integer programming problems. To see this, note that all bounded integer variables can be represented by a corresponding set of binary variables, and then every binary variable  $y \in \{0, 1\}$  can be equivalently represented by its continuous extension  $y \in [0, 1]$  and the nonconvex constraint  $y(1 - y) \leq 0$ . Of course, this direct reformulation is not (practically) suitable for many pure



or mixed integer optimization models. However, it certainly shows the generality of the CGO model. Other frequently studied special cases of the CGO model include e.g., the following (simply listed in alphabetical order, without establishing here their hierarchy):

- Concave programming: a concave objective minimized over a convex set
- Differential convex (DC) problems: in these all model functions  $f$  and  $g_i$ ,  $i = 1, \dots, m$  can be expressed as the difference of two suitable convex functions
- Fractional programming problems, where the objective is the ratio of two functions and the constraints are typically linear
- General (indefinite) quadratic programs, where the constraints are linear and the objective is a quadratic function, which is neither convex or concave, because its Hessian matrix is indefinite
- General quadratic programs under general quadratic constraints
- Linear or nonlinear complementarity problems: the model (constraint) functions satisfy a complementarity condition
- Lipschitz models: all model functions satisfy a corresponding Lipschitz-continuity condition
- Minimax problems: the model objective is a minimax function
- Multiplicative programming problems, where some the model functions are the product of several (typically convex) functions
- Nonsmooth problems, where the objective and/or constraint functions do not have continuous first partial derivatives everywhere, and some model functions are not convex either
- Polynomial optimization problems: the objective and constraint functions can all be general polynomials.

Let us observe next that if the feasible set  $D$  is non-empty, then the key analytical assumptions postulated above guarantee that the optimal solution set  $X^*$  of the CGO model is nonempty. This result is directly followed by the classical Bolzano-Weierstrass theorem, which states the existence of the global minimizer point—or, in general, a set of such points—of a continuous function over a nonempty, bounded, and closed (compact) set. For reasons of better numerical tractability, the following additional requirements are also often postulated:

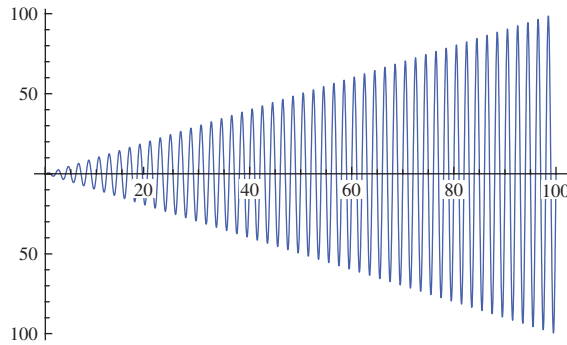
- $D$  is a full-dimensional subset (“body”) in  $\mathbf{R}^n$ ;
- the set of globally optimal solutions to (1)–(2) is at most countable;
- $f$  and  $g$  (component-wise) are Lipschitz-continuous functions on  $[l, u]$ .

Without going into technical details, let us remark that the first of these assumptions (the set  $D$  is the closure of its nonempty interior) makes algorithmic search easier—or at all possible—within  $D$ . The second assumption supports theoretical convergence arguments based on sample point sequences. (Note in passing that in most well-posed practical GO problems the set of global optimizers consists either of a single point  $x^*$  or at most of several points.) The third assumption is a sufficient condition for estimating  $f^* = f(x^*)$  on the basis of a finite set of algorithmically generated feasible search points. (Recall that the real-valued function  $h$  is Lipschitz-continuous on its domain of definition  $D \subset \mathbf{R}^n$ , if  $|h(x_1) - h(x_2)| \leq L\|x_1 - x_2\|$  holds for all pairs  $x_1 \in D$ ,  $x_2 \in D$ ; here  $L = L(D, h)$  is a suitable Lipschitz-constant of  $h$  on the set  $D$ .) We emphasize that the knowledge of the smallest suitable Lipschitz-constant for each model function is not required, and in practice such information is typically unavailable. At the same time, all models defined by continuously differentiable functions  $f$  and  $g$  belong to the CGO or even to the Lipschitz model class.

The above remarks already imply that the CGO model covers a very broad range of optimization problems. As a consequence of this generality, it includes also many model instances that are difficult to solve numerically. To illustrate this point, let us first consider a merely one-dimensional, box-constrained GO model

$$\min x(\sin \pi x) \quad 0 \leq x \leq 10. \quad (3)$$

FIGURE 1. The objective function in Model (3), when the RHS bound is 100.



This simple model should not cause too much grief to a genuine global scope optimization method, even if it is used with its default option or parameter settings. The (approximate, numerical) global solution of (3) is

$$x^* \approx 9.51064946978384995, \quad f^* \approx -9.50532721836634131.$$

This solution (found by one of our software packages that will be briefly discussed later on) can be visually verified, by plotting the objective function in (3) within the stated bounds  $[0, 10]$ . Consider now gradually changing the right-hand side (RHS) bound 10 to 100, to 1,000, to 10,000, and so on. Although in this simple example we do know that the global solution will be located “somewhere near” to the actual RHS bound, most GO software implementations will have an increasingly hard time to find the global solution when used “blindly” (without prior model-structure examinations, and using solver default settings). Figure 1 shows the objective function of (3) in the variable range  $0 \leq x \leq 100$ .

The global solution of the model depicted by Figure 1 is

$$x^* \approx 99.5010182888729702, \quad f^* \approx -99.5005091469001712.$$

To emphasize the nontrivial nature of these relatively simple examples, let us mention that the built-in GO function of a state-of-the-art software package that one of us (Pintér) is using misses the global solution by a wide margin, even in the much simpler case where the RHS bound equals 10 (at least when the function is used in its default mode). This comment could (and, in fact, does) apply also to some other current GO software packages.

Model complexity could increase at an exponential rate as the model size (expressed by the number of variables and the number of constraints) increases. To illustrate this point visually by another example, Figure 2 shows the objective function in Model (4) that is a very simple generalization of (3):

$$\min x(\sin \pi x) + y(\sin \pi y) \quad 0 \leq x \leq 10, \quad 0 \leq y \leq 10. \quad (4)$$

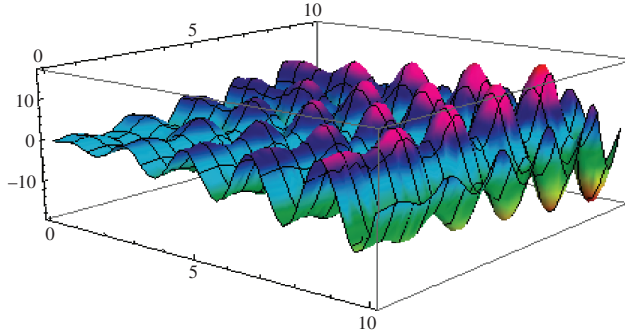
The approximate global solution of this model (obtained automatically by the same package that was used earlier) is

$$x^* \approx 9.51064947610772293, \quad y^* \approx 9.51064945678867701, \quad f^* \approx -19.0106544367326720.$$

In this simple case one can still directly verify (using analytical arguments) that the numerical solution found is a fairly close estimate of the “true” global solution: however, such direct verification is generally not possible.

The presented simple (low-dimensional, box-constrained) models already indicate that CGO models—for instance, further variants and/or higher dimensional extensions of

FIGURE 2. The objective function in Model (4).



Model (4), perhaps with added nonlinear constraints—could rapidly become difficult to solve. One should also point out that a direct analytical solution approach to CGO models is viable only in very special cases, because in general one should investigate all Kuhn-Tucker points—minimizers, maximizers, and saddle points—of the model. (Think of carrying out such a formal analysis e.g., for Model (4), or for its various, possibly far more complicated 10-, 100-, 1,000-dimensional extensions.)

Arguably, not all GO models are as difficult as indicated by Figures 1 and 2; at the same time, models could be numerically far more difficult to handle than these examples. As a rule, we do not have the possibility to directly inspect, visualize, or precisely estimate the overall numerical difficulty of a complicated GO model. Let us recall here the practical problem of automatically optimizing the parameters of a descriptive system model that has been developed by someone else. The descriptive model may be confidential, or just known to be complex; it could be presented to the optimization engine as a compiled (object, library, or similar “closed”) software module. In such situations, direct model inspection and structure verification are not possible.

Nonlinear models with less “dramatic” difficulty, but in (perhaps much) higher dimensions also require GO. For instance, in various engineering design contexts optimization models with tens, hundreds, or thousands of variables and constraints are developed. In similar cases, even an approximately completed, but genuinely global scope search strategy may—and typically will—yield better results than the most sophisticated local-search approach “started from the wrong valley”... This fact has significantly motivated research to develop viable GO approaches.

## 2. Global Optimization Strategies

As of today, well over a hundred textbooks and a number of websites are devoted to GO. The most important GO model types and solution approaches are discussed in the *Handbook of Global Optimization* volumes, edited by Horst and Pardalos [28], and by Pardalos and Romeijn [61]. One should also mention here the flagship *Journal of Global Optimization* (published since 1991). The topical website of Neumaier [54] also provides useful discussions with numerous links to other information sources. The very concise review of GO strategies presented here draws on these sources, as well as on the more detailed expositions in Edgar et al. [13] and Pintér [63, 65, 66].

As mentioned above, the general CGO model statement covers also classes of far more specific models. If a GO problem belongs to one of these more specific model classes, then specialized solution methods designed for such models will be usually more effective than a general-purpose approach. At the same time, specialized methods will often fail when applied outside of their intended scope.

To provide clear guidelines, we can classify GO methods into two primary categories: exact and heuristic. Exact methods possess theoretically established—deterministic or

stochastic—global convergence properties. That is, if such a method could be carried out completely as an infinite iterative process, then the generated limit point(s) would belong to the set of global solutions  $X^*$ . (For a single global solution  $x^*$ , this would be the only limit point.) In the case of stochastic GO methods, the above statement is valid “only” with probability one (w.p.1). In practice—after a finite number of algorithmic search steps—one can only expect a numerically validated or estimated (deterministic or stochastic) lower bound for the global optimum value  $z^* = f(x^*)$ , as well as a best feasible (or nearest-to-feasible) global solution estimate. We wish to emphasize that to produce such estimates is not a trivial task, even for implementations of theoretically well-established algorithms. There is no GO method—and never will be one—that can solve “all possible” CGO models with a given number of variables to an arbitrarily chosen precision (in terms of the argument  $x^*$ ), within a given time frame, or within a preset model function evaluation count. It is not difficult to fabricate CGO models that could make life hard for any deterministic or stochastic solver with its preset (default) options, in the sense outlined above.

Heuristic methods are often based on plausibility arguments or natural analogies, but such approaches do not possess similar convergence guarantees to those of exact methods. At the same time, they may provide good quality solutions in many difficult (both continuous and discrete) GO problems, within an acceptable timeframe—assuming that the method in question suits well the specific model type solved. Because heuristic strategies are often based on some generic meta-heuristics, overly optimistic claims regarding their “universal” efficiency are not always supported by results in solving truly difficult—especially nonlinearly constrained—GO models. In addition, heuristic meta-strategies are often more difficult to adjust to new model types than some of the solver implementations based on theoretically exact algorithms.

## 2.1. Exact Methods

In this section, we present a very brief summary of the most-often used exact GO strategies. For more details, please consult the references.

“Naïve” or “passive” approaches (such as deterministic grid search and pure random search) are obviously convergent in a deterministic sense or w.p.1, but in general these are also hopelessly inefficient as the problem size and/or model complexity increases.

Branch-and-bound methods are very general approaches that subsume interval arithmetic-based strategies, as well as customized approaches e.g., for Lipschitz GO, and for certain more special classes such as concave minimization or DC models. Such methods can also be applied to constraint satisfaction problems and to (general) pure and mixed integer programming.

Homotopy (path following, deformation, continuation, trajectory, and related other) methods are aimed at finding the set of global solutions in smooth GO models.

Examples of implicit enumeration techniques are vertex enumeration in concave minimization models and generic dynamic programming in the context of combinatorial optimization.

Outer approximation methods (including various cutting-plane and relaxation algorithms) solve a sequence of increasingly tight approximations to an optimization problem, where the feasible set of the approximating problem always contains the original feasible region.

Stochastically convergent sequential sampling methods include adaptive random searches, single- and multi-start methods, Bayesian search strategies, and their combinations.

For detailed expositions related to exact GO techniques—in addition to the *Handbooks* [28, 61] mentioned earlier—consult for example Horst and Tuy [29], Kearfott [35], Neumaier [53], Nowak [56], Pintér [63], and Tawarmalani and Sahinidis [81]. Regarding exact stochastic GO strategies, consult e.g., Boender and Romeijn [6], Edgar et al. [13], Pintér [63], Zabinsky [94], and Zhigljavsky [95].

## 2.2. Heuristic Methods

Similarly to the previous section, now we list a few key heuristic GO strategies. For more details, please consult the references.

Ant colony optimization is based on individual search steps and “ant-like” interaction (dynamic communication) between search agents.

Basin-hopping strategies are based on a sequence of perturbed local searches, in an attempt to find improving optima.

Convex underestimation attempts are based on a limited sampling effort that is used to estimate a postulated (approximate) convex objective function model.

Evolutionary search methods model the behavioral linkage among an adaptively changing set of candidate solutions (“parents” and their “children,” in a sequence of “generations”).

Genetic algorithms emulate specific genetic operations (selection, crossover, mutation) as these are observed in nature, similarly to evolutionary methods.

Greedy adaptive search strategies—a meta-heuristics often used in combinatorial optimization—quickly construct “promising” initial solutions, which are then refined by a suitable local optimization procedure.

Memetic algorithms are inspired by analogies to cultural (as opposed to natural) evolution.

Neural networks are based on a model of the parallel architecture of the brain.

Response surface methods (directed sampling techniques) are often used in handling expensive “black-box” optimization models by postulating and then gradually adapting a surrogate function model.

Scatter search is similar in its algorithmic structure to ant colony, genetic, and evolutionary searches, but without their “biological inspiration.”

Simulated annealing methods are based on the analogy of cooling crystal structures that will attain a (low-energy level, stable) physical equilibrium state.

Tabu search forbids or penalizes search moves that take the current solution in the next few iterations to points with specific properties (including previously visited points or ones that are “too similar” to these) in the solution space.

Tunneling strategies, filled function methods, and other similar strategies attempt to find an improving sequence of local optima, by gradually modifying the objective function to escape from the solutions found.

In addition to the mentioned topical GO books, we refer here to several works that mostly discuss combinatorial (but also some continuous) GO models and heuristic strategies. For discussions of theory and applications, consult for example Edgar et al. [13], Ferreira [14], Glover and Laguna [23], Goldberg [24], Jones and Pevzner [32], Michalewicz [48], Osman and Kelly [57], Rothlauf [75], and Voss et al. [87]. It is worth pointing out that Rudolph [76] discusses the typically missing theoretical foundations for evolutionary algorithms, including stochastic convergence studies: The underlying key convergence results for adaptive stochastic search methods are discussed in Pintér [63]. The topical chapters in Pardalos and Resende [60] also offer expositions related to both exact and heuristic GO approaches.

To conclude this very concise review of GO strategies, let us emphasize again that numerical GO can be tremendously difficult. Therefore it can be prudent practice to try several—perhaps (or even preferably) radically different—algorithmic search approaches to tackle GO problems, whenever this is possible. To do this effectively, one needs ready-to-use model development and optimization software tools.

## 3. Global Optimization in Modeling Environments

Advances in model development techniques, algorithms, software, and computer hardware technology have led to growing interest towards optimization (as well as more general-purpose) modeling environments. For detailed discussions of optimization modeling systems consult e.g., the topical *Annals of Operations Research* volumes edited by Maros and

Mitra [44], Maros et al. [45], Vladimirov et al. [86], Coullard et al. [12], as well as the volumes edited by Voss and Woodruff [88] and by Kallrath [33]. Additional information is provided e.g., by Moré and Wright [51], as well as by the websites of Fourer [17], Mittelmann [49], and Neumaier [54] with numerous further links. Here we will restrict the discussion to currently available GO solver engines; in this context, see also Liberti and Maculan [38].

There exist (mostly C or Fortran) compiler platform-based GO software packages, equipped with more or less friendly model-development functionality. In principle, all such solvers can be linked to modeling languages and to other integrated development environments discussed below. Here we will highlight only those packages that are actually linked to one or several model development environments (to our best knowledge).

Prominent examples of widely used modeling systems that are focused on optimization include AIMMS (Paragon Decision Technology [59]), AMPL (Fourer et al. [18]), the Excel Premium Solver Platform (Frontline Systems [20]), GAMS (Brooke et al. [8]), ILOG [30], the LINDO Solver Suite (LINDO Systems [39]), MPL (Maximal Software [47]), and TOMLAB [83]. For up-to-date information, contact the developers of these systems and/or visit their website.

There is also notable development in relation to integrated scientific and technical computing (ISTC) systems such as Maple (Maplesoft [43]), *Mathematica* (Wolfram Research [92]), Mathcad (Mathsoft [46]), and MATLAB (The MathWorks [82]). From among the many hundreds of books discussing ISTC systems, we mention here only a few works by Birkeland [5], Bhatti [3], Lopez [40], Moler [50], Parlar [62], Trott [84], Wilson et al. [89], Wolfram [91], and Wright [93]. In addition to their overall (modeling, computing, graphical, documentation, etc.) functionality, ISTC systems offer also a range of optimization features, either as built-in functions or as add-on products.

The modeling environments listed above are aimed (together) at meeting the needs of a broad and diverse clientele. Client categories include educational users (instructors and students); research scientists, engineers, consultants, and other practitioners from various fields; optimization experts, software application developers, and other “power users.” The pros and cons of the individual software products—in terms of hardware- and software-platform demands, ease of use, model prototyping options, detailed code development and maintenance features, optimization model checking and processing tools, availability of solver options and other auxiliary tools, program execution speed, overall level of system integration, quality of related documentation and support, customization options, and communication with end users—make the corresponding model development and solution approaches more or less attractive for the various user groups.

Given the massive amount of topical information, which are the currently available platform and solver engine choices for the GO academic, researcher, or practitioner? The now over a decade old software review (Pintér [64]; available also at the website of Mittelmann [49]) lists a few dozens of GO software projects and products, including references to several websites with further software collections. Neumaier’s [54] webpage currently lists more than one hundred GO software development projects. These include general purpose solvers, as well as application-specific products. (As a side note, quite a few of the links—both in Pintér’s and Neumaier’s project lists—seem to be obsolete, or the related websites have changed.)

The user’s preference obviously depends on many (often contradicting) factors. A key question is whether one prefers to use “free” (that is, noncommercial, research, or even open source) code, or looks for a “ready-to-use,” professionally supported commercial product. There is a significant inventory of freely available solvers, although the quality of the underlying methods, the actual implementations, and their documentation varies widely. (Of course, this remark could well apply also to commercial products.) Instead of trying to impose personal judgment on any of the products mentioned or just referred to here, the reader is encouraged to do some web browsing and experimentation, as his/her time

and resources allow. Both Mittelman [49] and Neumaier [54] provide far more extensive information on noncommercial—as opposed to commercial—systems. To supplement that information, we shall mention here software products that are part of commercial systems, typically as an add-on option, but in some cases as a built-in option. We only list currently available products that are explicitly targeted toward GO, as advertised by the websites of the listed companies. For this reason, nonlinear (but local scope) solvers are not listed here; furthermore, we will not list modeling environments that currently have no global solver options.

*AIMMS*, by Paragon Decision Technology ([www.aimms.com](http://www.aimms.com)): the BARON and LGO global solver engines are offered with this modeling system as add-on options.

*GAMS*, by the GAMS Development Corporation ([www.gams.com](http://www.gams.com)): currently, AlphaECP, BARON, Bonmin, DICOPT, LGO, LINGOGlobal, MSNLP, OQNLP, and SBB are offered as GO solver options.

*LINDO*, by LINDO Systems ([www.lindo.com](http://www.lindo.com)): both the LINGO modeling environment and What'sBest! (the company's spreadsheet solver) have built-in global solver functionality.

*Maple*, by Maplesoft ([www.maplesoft.com](http://www.maplesoft.com)) offers the Global Optimization Toolbox (GOT) as an add-on product. The GOT is based on the LGO solver suite linked to Maple; see Maplesoft [42].

*Mathematica*, by Wolfram Research ([www.wolfram.com](http://www.wolfram.com)) has a built-in function (called NMinimize) for numerical GO. In addition, there are several third-party GO packages that can be directly linked to Mathematica. These are the native Mathematica packages Global Optimization, MathOptimizer, and MathOptimizer Professional (the LGO solver suite with a link to Mathematica).

*MPL*, by Maximal Software ([www.maximal-usa.com](http://www.maximal-usa.com)): LGO is offered as an optional solver engine.

*Premium Solver Platform (PSP) for Excel*, by Frontline Systems ([www.solver.com](http://www.solver.com)). The developers of the PSP offer a global presolver option that can be used with several of the available local optimization engines. These currently include LSGRG, LSSQP, and KNITRO. Frontline Systems also offers (as genuine global solvers) an Interval Global Solver, an Evolutionary Solver, and OptQuest.

*TOMLAB*, by TOMLAB Optimization AB ([www.tomopt.com](http://www.tomopt.com)) is an optimization platform for solving MATLAB models. The TOMLAB global solvers currently include CGO, LGO, MINLP, and OQNLP. Note that MATLAB's own Genetic Algorithm and Direct Search Toolboxes also have heuristic global solver capabilities.

Concluding this brief review, let us mention that among the listed solvers LSGRG, MSNLP, and OQNLP are developed by Lasdon, and that LGO is developed by Pintér—in both cases in cooperation with a number of developer partners on platform-specific implementations. These systems have been discussed elsewhere in details. Regarding LSGRG, MSNLP, and OQNLP, consult e.g., Lasdon et al. [37], Edgar et al. [13], and Ugray et al. [85]. LGO and several of its implementations are discussed e.g., in Pintér [63, 65, 67–70]; Pintér and Kampas [71, 72]; and Pintér et al. [73]. The illustrative examples presented in §1 have been solved using the GOT implementation of LGO; many further numerical examples can be found in the works referred to above.

## 4. Applications

Global optimization has gradually become an established field within operations research. GO methods and software are also increasingly applied in various research and practical contexts. The currently available professional software implementations can be (have been) used to solve difficult GO models with tens, hundreds, and sometimes even thousands of variables and constraints. Recall, however, the potential numerical difficulty of GO model instances: if one is interested in a guaranteed and precise solution, then the necessary

runtimes could become minutes, hours, days, or more—even on today’s high-performance computers. Of course, one can expect further speed-up due to both algorithmic improvements and to progress in hardware/software technology, but the theoretically exponential “curse of dimensionality” associated with the subject of computational GO will always be there.

In the most general terms, GO technology is well-suited to analyze and solve nonlinear models arising in applied mathematics, physics, chemistry, biology, medical and pharmaceutical studies, environmental sciences, engineering, econometrics, and financial modeling. To illustrate the wide range of GO application areas, below we will list a number of these.

The collection of test problems by Floudas et al. [16] includes models from the following areas:

- (chemical) batch plant design under uncertainty
- conformational problems in clusters of atoms and molecules
- dynamic optimization problems in parameter estimation
- homogeneous azeotropic separation system
- network synthesis
- optimal control problems
- parameter estimation and data reconciliation
- pump network synthesis
- robust stability analysis
- trim loss minimization.

The edited volume Pintér [67] includes detailed engineering and scientific case studies from the following fields:

- agro-ecosystem management
- assembly line design
- bioinformatics
- biophysics
- cancer therapy planning
- cellular mobile network design
- chemical process optimization
- chemical product design
- composite structure (material) design
- computational modeling of atomic and molecular structures
- controller design for induction motors
- electrical engineering design
- laser design
- learning in neural nets
- mechanical engineering design
- numerical solution of equations
- optimization of feeding strategies in animal husbandry
- radiotherapy planning
- robot design
- satellite data analysis
- solving the inverse position problem in kinematics.

Many other important areas of GO applications are discussed e.g., by Corliss and Kearfott [11], Edgar et al. [13], Floudas [15], Grossmann [25], Kampas and Pintér [34], Nowak [56], Papalambros and Wilde [58], Pintér [65, 66, 68, 70], Schittkowski [77], Tawarmalani and Sahinidis [81], and Zabinsky [94]. These works include numerical examples and case studies e.g., from the application areas:

- “black-box” systems optimization
- combination of negotiated expert opinions (forecasts, assessments, etc.)



- constraint system satisfiability problems
- data classification, pattern recognition
- dynamic population management
- environmental (water quality and other) management
- facility location
- financial modeling (stochastic process model calibration)
- financial portfolio management
- finite element modeling and optimization in sonar equipment design
- industrial (object) design
- layout design
- mechanical engineering design
- model calibration (nonlinear regression)
- object packings and their industrial applications
- optical design
- product (base material mixture) design
- prototyping in engineering R&D
- risk analysis and control in various environmental management contexts
- robotics design
- supply chain reliability optimization
- statistical modeling
- systems of nonlinear equations and inequalities
- telecommunications.

The websites of Mittelman and Neumaier, as well as the sites of the software developers listed earlier also offer collections of GO test examples, practically motivated nonlinear optimization models, and detailed case studies.

## 5. Conclusions

GO is a subject of increasing practical interest. This fact is indicated also by GO software implementations and by a rapidly growing range of applications. In this chapter we have highlighted and briefly discussed some of these developments. In spite of remarkable progress, GO remains a field of extreme numerical challenges, not only when considering “all possible” GO models, but also in practical attempts to handle complex, sizeable problems within an acceptable timeframe. In spite of this caveat, the practice of GO is expected to gain further territory at a rapid pace. We welcome feedback regarding suggested development directions, new application areas and test challenges.

## References

- [1] R. Aris. *Mathematical Modeling: A Chemical Engineer's Perspective*. Academic Press, San Diego, CA, 1999.
- [2] M. S. Bazaraa, H. D. Sherali, and C. M. Shetty. *Nonlinear Programming: Theory and Algorithms*. Wiley, New York, 1993.
- [3] M. A. Bhatti. *Practical Optimization Methods with Mathematica Applications*. Springer-Verlag, New York, 2000.
- [4] D. P. Bertsekas. *Nonlinear Programming*, 2nd ed. Athena Scientific, Cambridge, MA, 1999.
- [5] B. Birkeland. *Mathematics with Mathcad*. Studentlitteratur/Chartwell Bratt, Lund, Sweden, 1997.
- [6] C. G. E. Boender and H. E. Romeijn. Stochastic methods. R. Horst and P. M. Pardalos, eds. *Handbook of Global Optimization*, Vol. 1. Kluwer Academic Publishers, Dordrecht, The Netherlands, 829–869, 1995.
- [7] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004.

- [8] A. Brooke, D. Kendrick, and A. Meeraus. *GAMS: A User's Guide*. The Scientific Press, Redwood City, CA. (Revised versions are available from the GAMS Corporation.) <http://www.gams.com>, 1988.
- [9] J. L. Casti. *Searching for Certainty*. Morrow & Co., New York, 1990.
- [10] E. K. P. Chong and S. H. Zak. *An Introduction to Optimization*, 2nd ed. Wiley, New York, 2001.
- [11] G. F. Corliss and R. B. Kearfott. Rigorous global search: Industrial applications. T. Csendes, ed. *Developments in Reliable Computing*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1–16, 1999.
- [12] C. Coullard, R. Fourer, and J. H. Owen, eds. *Annals of Operations Research, Vol. 104: Special Issue on Modeling Languages and Systems*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2001.
- [13] T. F. Edgar, D. M. Himmelblau, and L. S. Lasdon. *Optimization of Chemical Processes*, 2nd ed. McGraw-Hill, New York, 2001.
- [14] C. Ferreira. *Gene Expression Programming: Mathematical Modeling by an Artificial Intelligence*. Angra do Heroísmo, Portugal, 2002.
- [15] C. A. Floudas. Deterministic global optimization and its applications. P. M. Pardalos and M. G. C. Resende, eds. *Handbook of Applied Optimization*. Oxford University Press, Oxford, UK, 311–336, 2002.
- [16] C. A. Floudas, P. M. Pardalos, C. Adjiman, W. R. Esposito, Z. H. Gumus, S. T. Harding, J. L. Klepeis, C. A. Meyer, and C. A. Schweiger. *Handbook of Test Problems in Local and Global Optimization*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999.
- [17] R. Fourer. *Nonlinear Programming Frequently Asked Questions*. Optimization Technology Center of Northwestern University and Argonne National Laboratory. <http://www-unix.mcs.anl.gov/otc/Guide/faq/nonlinear-programming-faq.html>, 2007.
- [18] R. Fourer, D. M. Gay, and B. W. Kernighan. *AMPL—A Modeling Language for Mathematical Programming*. The Scientific Press, Redwood City, CA. (Reprinted by Boyd and Fraser, Danvers, MA, 1996.) <http://www.ampl.com>, 1993.
- [19] P. Fritzson. *Principles of Object-Oriented Modeling and Simulation with Modelica 2.1*. IEEE Press, Wiley-Interscience, Piscataway, NJ, 2004.
- [20] Frontline Systems. *Premium Solver Platform—Solver Engines. User Guide*. Frontline Systems, Inc., Incline Village, NV. <http://www.solver.com>, 2007.
- [21] D. Y. Gao, R. W. Ogden, and G. E. Stavroulakis, eds. *Nonsmooth/Nonconvex Mechanics: Modeling, Analysis and Numerical Methods*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2001.
- [22] N. Gershenfeld. *The Nature of Mathematical Modeling*. Cambridge University Press, Cambridge, UK, 1999.
- [23] F. Glover and M. Laguna. *Tabu Search*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1997.
- [24] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, Reading, MA, 1989.
- [25] I. E. Grossmann, ed. *Global Optimization in Engineering Design*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996.
- [26] P. E. Hansen and S. E. Jørgensen, eds. *Introduction to Environmental Management*. Elsevier, Amsterdam, The Netherlands, 1991.
- [27] F. J. Hillier, and G. J. Lieberman. *Introduction to Operations Research*, 8th ed. McGraw-Hill, New York, 2005.
- [28] R. Horst and P. M. Pardalos, eds. *Handbook of Global Optimization*, Vol. 1. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1995.
- [29] R. Horst and H. Tuy. *Global Optimization—Deterministic Approaches*, 3rd ed. Springer, Berlin, Germany, 1996.
- [30] ILOG. ILOG OPL Studio and Solver Suite. <http://www.ilog.com>, 2007.
- [31] C. Jacob. *Illustrating Evolutionary Computation with Mathematica*. Morgan Kaufmann Publishers, San Francisco, CA, 2001.
- [32] N. C. Jones and P. A. Pevzner. *An Introduction to Bioinformatics Algorithms*. The MIT Press, Cambridge, MA, 2004.

- [33] J. Kallrath, ed. *Modeling Languages in Mathematical Optimization*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2004.
- [34] F. J. Kampas and J. D. Pintér. *Optimization with Mathematica*. Forthcoming.
- [35] R. B. Kearfott. *Rigorous Global Search: Continuous Problems*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996.
- [36] L. S. Lasdon. *Optimization Theory for Large Systems*, 2nd ed. Dover Publications, Mineola, NY, 2002.
- [37] L. Lasdon, A. Waren, A. Jain, and M. Ratner. Design and testing of a generalized reduced gradient code for nonlinear programming. *ACM Transactions on Mathematical Software* 4(1):34–50, 1978.
- [38] Liberti and Maculan, eds. *Global Optimization: From Theory to Implementation*. Springer Science + Business Media, New York, 2005.
- [39] LINDO Systems. *Solver Suite*. LINDO Systems, Inc., Chicago, IL. <http://www.lindo.com>, 1996.
- [40] R. J. Lopez, *Advanced Engineering Mathematics with Maple*. (Electronic book edition.) Maplesoft Inc., Waterloo, Ontario, Canada. <http://www.maplesoft.com/products/ebooks/AEM/>, 2005.
- [41] B. B. Mandelbrot. *The Fractal Geometry of Nature*. Freeman & Co., New York, 1983.
- [42] Maplesoft. *Global Optimization Toolbox for Maple*. Maplesoft Inc., Waterloo, Ontario, Canada. <http://www.maplesoft.com/products/toolboxes/globaloptimization/>, 2004.
- [43] Maplesoft. *Maple*. Maplesoft Inc., Waterloo, Ontario, Canada. <http://www.maplesoft.com>, 2007.
- [44] I. Maros and G. Mitra, eds. *Annals of Operations Research, Vol. 58: Applied Mathematical Programming and Modeling II (APMOD 93)*. J. C. Baltzer AG Science Publishers, Basel, Switzerland, 1995.
- [45] I. Maros, G. Mitra, and A. Sciomachen, eds. *Annals of Operations Research, Vol. 81: Applied Mathematical Programming and Modeling III (APMOD 95)*. J. C. Baltzer AG Science Publishers, Basel, Switzerland, 1997.
- [46] Mathsoft. *Mathcad*. Mathsoft Engineering & Education Inc., Cambridge, MA, 2007.
- [47] Maximal Software. *MPL Modeling System*. Maximal Software Inc., Arlington, VA. <http://www.maximal-usa.com>, 2007.
- [48] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs*, 3rd ed. Springer, New York, 1996.
- [49] H. D. Mittelmann. *Decision Tree for Optimization Software*. <http://plato.la.asu.edu/guide.html>, 2007.
- [50] C. B. Moler. *Numerical Computing with MATLAB*. SIAM, Philadelphia, PA, 2004.
- [51] J. J. Moré and S. J. Wright. *Optimization Software Guide*. SIAM, Philadelphia, PA, 1993.
- [52] J. D. Murray. *Mathematical Biology*. Springer-Verlag, Berlin, Germany, 1983.
- [53] A. Neumaier. Complete search in continuous global optimization and constraint satisfaction. A. Iserles, ed. *Acta Numerica 2004*, Cambridge University Press, Cambridge, UK, 271–369, 2004.
- [54] A. Neumaier. *Global Optimization*. <http://www.mat.univie.ac.at/~neum/glopt.html>, 2007.
- [55] J. Nocedal and S. J. Wright. *Numerical Optimization*. 2nd ed. Springer Science + Business Media, New York, 2006.
- [56] I. Nowak. *Relaxation and Decomposition Methods for Mixed Integer Nonlinear Programming*. Birkhäuser, Basel, Switzerland, 2005.
- [57] I. H. Osman and J. P. Kelly, eds. *Meta-Heuristics: Theory and Applications*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996.
- [58] P. Y. Papalambros and D. J. Wilde. *Principles of Optimal Design*. Cambridge University Press, Cambridge, UK, 2000.
- [59] Paragon Decision Technology. AIMMS. Paragon Decision Technology BV, Haarlem, The Netherlands. <http://www.aimms.com>, 2007.
- [60] P. M. Pardalos and M. G. C. Resende, eds. *Handbook of Applied Optimization*. Oxford University Press, Oxford, UK, 2002.
- [61] P. M. Pardalos and H. E. Romeijn, eds. *Handbook of Global Optimization*, Vol. 2. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.
- [62] M. Parlar. *Interactive Operations Research with Maple*. Birkhäuser, Boston, MA, 2000.

- [63] J. D. Pintér. *Global Optimization in Action*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1996.
- [64] J. D. Pintér. Continuous global optimization software: A brief review. *Optima* 52:1–8. <http://plato.la.asu.edu/gom.html>. 1996.
- [65] J. D. Pintér. *Computational Global Optimization in Nonlinear Systems*. Lionheart Publishing Inc., Atlanta, GA, 2001.
- [66] J. D. Pintér. Global optimization: Software, test problems, and applications. P. M. Pardalos and H. E. Romeijn, eds. *Handbook of Global Optimization*, Vol. 2. Kluwer Academic Publishers, Dordrecht, The Netherlands, 515–569, 2002.
- [67] J. D. Pintér, ed. *Global Optimization—Scientific and Engineering Case Studies*. Springer Science + Business Media, New York, 2006.
- [68] J. D. Pintér. *Global Optimization with Maple: An Introduction with Illustrative Examples*. Pintér Consulting Services Inc., Halifax, Nova Scotia, Canada and Maplesoft, Waterloo, Ontario, Canada, 2006.
- [69] J. D. Pintér. Nonlinear optimization with GAMS/LGO. *Journal of Global Optimization* 38:79–101, 2007.
- [70] J. D. Pintér. *Applied Nonlinear Optimization in Modeling Environments*. CRC Press, Boca Raton, FL. Forthcoming.
- [71] J. D. Pintér, and F. J. Kampas. Model development and optimization with Mathematica. B. Golden, S. Raghavan, and E. Wasil, eds. *Proceedings of the 2005 INFORMS Computing Society Conference* (Annapolis, MD, January 2005), Springer Science + Business Media, New York, 285–302, 2005.
- [72] J. D. Pintér and F. J. Kampas. Nonlinear optimization in *Mathematica* with *MathOptimizer Professional*. *Mathematica in Education and Research* 10:1–18, 2005.
- [73] J. D. Pintér, D. Linder, and P. Chin. Global Optimization Toolbox for Maple: An introduction with illustrative applications. *Optimization Methods and Software* 21(4):565–582, 2006.
- [74] L. G. Rich. *Environmental Systems Engineering*. McGraw-Hill, Tokyo, Japan, 1973.
- [75] F. Rothlauf. *Representations for Genetic and Evolutionary Algorithms*. Physica-Verlag, Heidelberg, Germany, 2002.
- [76] G. Rudolph. *Convergence Properties of Evolutionary Algorithms*. Verlag Dr. Kovac, Hamburg, Germany, 1997.
- [77] K. Schittkowski. *Numerical Data Fitting in Dynamical Systems*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.
- [78] M. Schroeder. *Fractals, Chaos, Power Laws*. Freeman & Co., New York, 1991.
- [79] I. Stewart. *Nature's Numbers*. Basic Books/Harper and Collins, New York, 1995.
- [80] S. Stojanovic. *Computational Financial Mathematics Using Mathematica*. Birkhäuser, Boston, MA, 2003.
- [81] M. Tawarmalani and N. V. Sahinidis. *Convexification and Global Optimization in Continuous and Mixed-Integer Nonlinear Programming*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.
- [82] The MathWorks. *MATLAB*. The MathWorks, Inc., Natick, MA. <http://www.mathworks.com>, 2007.
- [83] TOMLAB Optimization. TOMLAB. TOMLAB Optimization AB, Västerås, Sweden. <http://www.tomopt.com>, 2007.
- [84] M. Trott. *The Mathematica GuideBooks*, Vols. 1–4. Springer Science + Business Media, New York, 2004.
- [85] Zs. Ugray, L. Lasdon, J. Plummer, F. Glover, J. Kelly, and R. Marti. Scatter search and local NLP solvers: A multistart framework for global optimization. McCombs Research Paper Series No. IROM-07-06, University of Texas at Austin, Austin, TX, 2006.
- [86] H. Vladimirou, I. Maros, and G. Mitra, eds. *Annals of Operations Research, Vol. 99: Applied Mathematical Programming and Modeling IV (APMOD 98)*. J. C. Baltzer AG Science Publishers, Basel, Switzerland, 2000.
- [87] S. Voss, S. Martello, I. H. Osman, and C. Roucairol, eds. *Meta-Heuristics: Advances and Trends in Local Search Paradigms for Optimization*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1999.
- [88] S. Voss and D. L. Woodruff, eds. *Optimization Software Class Libraries*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.

- [89] H. B. Wilson, L. H. Turcotte, and D. Halpern. *Advanced Mathematics and Mechanics Applications Using MATLAB*, 3rd ed. Chapman and Hall/CRC Press, Boca Raton, FL, 2003.
- [90] S. Wolfram. *A New Kind of Science*. Wolfram Media, Champaign, IL, 2002.
- [91] S. Wolfram. *The Mathematica Book*, 4th ed. Wolfram Media, Champaign, IL, and Cambridge University Press, Cambridge, UK, 2003.
- [92] Wolfram Research. *Mathematica*. Wolfram Research Inc., Champaign, IL. <http://www.wolfram.com>, 2007.
- [93] F. Wright. *Computing with Maple*. Chapman and Hall/CRC Press, Boca Raton, FL, 2002.
- [94] Z. B. Zabinsky. *Stochastic Adaptive Search for Global Optimization*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2003.
- [95] A. A. Zhigljavsky. *Theory of Global Random Search*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1991.

# Coherent Approaches to Risk in Optimization Under Uncertainty

*R. Tyrrell Rockafellar*

Department of Industrial and Systems Engineering, University of Florida, Gainesville,  
Florida 32611, rtr@ise.ufl.edu

**Abstract** Decisions often need to be made before all the facts are in. A facility must be built to withstand storms, floods, or earthquakes of magnitudes that can only be guessed from historical records. A portfolio must be purchased in the face of only statistical knowledge, at best, about how markets will perform. In optimization, this implies that constraints may need to be envisioned in terms of safety margins instead of exact requirements. But what does that really mean in model formulation? What guidelines make sense, and what are the consequences for optimization structure and computation?

The idea of a coherent measure of risk in terms of surrogates for potential loss, which has been developed in recent years for applications in financial engineering, holds promise for a far wider range of applications in which the traditional approaches to uncertainty have been subject to criticism. The general ideas and main facts are presented here with the goal of facilitating their transfer to practical work in those areas.

**Keywords** optimization under uncertainty; safeguarding against risk; safety margins; measures of risk; measures of potential loss; measures of deviation; coherency; value-at-risk; conditional value-at-risk; probabilistic constraints; quantiles; risk envelopes; dual representations; stochastic programming

---

## 1. Introduction

In classical optimization based on deterministic modeling, a typical problem in  $n$  variables has the form

$$\text{minimize } c_0(x) \quad \text{over all } x \in S \text{ satisfying } c_i(x) \leq 0 \text{ for } i = 1, \dots, m, \quad (1.1)$$

where  $S$  is a subset of  $\mathbb{R}^n$  composed of vectors  $x = (x_1, \dots, x_n)$ , and each  $c_i$  is a function from  $S$  to  $\mathbb{R}$ . In the environment of uncertainty that dominates a vast range of applications, however, a serious difficulty arises in such a formulation. We can think of it as caused by parameter elements about which the optimizer (who wishes to solve the problem) has only incomplete information at the time  $x$  must be chosen. Decisions are then fraught with risk over their outcomes, and the way to respond may be puzzling.

The difficulty can be captured by supposing that instead of just  $c_i(x)$  we have  $c_i(x, \omega)$ , where  $\omega$  belongs to a set  $\Omega$  representing future states of knowledge. For instance,  $\Omega$  might be a subset of some parameter space  $\mathbb{R}^d$ , or merely a finite index set. The choice of an  $x \in S$  no longer produces specific numbers  $c_i(x)$ , as taken for granted in problem (1.1), but merely results in a collection of *functions* on  $\Omega$

$$c_i(x): \omega \rightarrow c_i(x, \omega) \quad \text{for } i = 0, 1, \dots, m. \quad (1.2)$$

How then should the constraints in the problem be construed? How should the objective be reinterpreted? In what ways should risk be taken into account? Safeguards may be needed

to protect against undesired outcomes, and safety margins may have to be introduced, but on what basis?

Various approaches, with their pros and cons, have commonly been followed and will be reviewed shortly as background for explaining more recent ideas. However, an important principle should be understood first: No conceptual distinction should be made between the treatment of the objective function  $c_0$  and the constraint functions  $c_1, \dots, c_m$  in these circumstances.

Behind the formulation of problem (1.1) there may have been a number of functions, all of interest in terms of keeping their values low. A somewhat arbitrary choice may have been made as to which one should be minimized subject to constraints on the others. Apart from this arbitrariness, well known device, in which an additional coordinate  $x_{n+1}$  is appended to  $x = (x_1, \dots, x_n)$ , can anyway always be used to convert (1.1) into an equivalent problem in which all the complications are put into the constraints and none are left in the objective, namely

$$\begin{aligned} & \text{minimize } x_{n+1} \quad \text{over all } (x, x_{n+1}) \in S \times \mathbb{R} \\ & \text{satisfying } \begin{cases} c_0(x) - x_{n+1} \leq 0 \\ c_i(x) \leq 0 \quad \text{for } i = 1, \dots, m. \end{cases} \quad \text{and} \end{aligned} \quad (1.3)$$

The challenges of uncertainty would be faced in the reformulated model with the elements  $\omega \in \Omega$  affecting only constraints.

This principle will help in seeing the modeling implications of different approaches to handling risk. Let us also note, before proceeding further, that in the notation  $c_i(x, \omega)$  some functions might only depend on a partial aspect of  $\omega$ , or perhaps not on  $\omega$  at all, although for our purposes, constraints not touched by uncertainty could be suppressed into the specification of the set  $S$ . Equations have been omitted as constraints because, if uncertain, they rarely make sense in this basic setting, and if certain, they could likewise be put into  $S$ .

Hereafter, we will think of  $\Omega$  as having the mathematical structure of a probability space with a probability measure  $P$  for comparing the likelihood of future states  $\omega$ . This is more a technical device rather than a philosophical statement. Perhaps there is true knowledge of probabilities, or a subjective belief in probabilities that should appropriately influence actions by the decision maker. But the theory to be described here will bring into consideration other probability measures as alternatives, and ways of suggesting at least what the probabilities of subdivisions of  $\Omega$  might be if a more complete knowledge is lacking. The designation  $P$  might therefore be just a way to begin.

By working with a probability measure  $P$  on  $\Omega$  we can interpret the functions  $c_i(x): \Omega \rightarrow \mathbb{R}$  as *random variables*. Any function  $X: \Omega \rightarrow \mathbb{R}$  induces a probability distribution on  $\mathbb{R}$  with cumulative distribution function  $F_X$  defined by taking  $F_X(z)$  to be the probability assigned by  $P$  to the set of  $\omega \in \Omega$  such that  $X(\omega) \leq z$ . (In the theory of probability spaces introducing a field of measurable sets in  $\Omega$ , and so forth, should be a concern. For this tutorial, however, such details are not considered.)

Integrals with respect to  $P$  will be written as expectations  $E$ . We will limit attention to random variables  $X$  for which  $E[X^2]$  is finite; such random variables make up a linear space that will be denoted here just by  $\mathcal{L}^2$ . Having  $X \in \mathcal{L}^2$  ensures that both the mean and standard deviation of  $X$ , namely

$$\mu(X) = EX \quad \text{and} \quad \sigma(X) = (E[(X - \mu(X))^2])^{1/2}$$

are well defined and finite. Here we will assume that all the specific random variables entering our picture through (1.2) for choices of  $x \in S$  belong to  $\mathcal{L}^2$ .

To recapitulate, in this framework uncertainty causes the numbers  $c_i(x)$  in the deterministic model (1.1) to be replaced by the random variables  $c_i(x)$  in (1.2). This casts doubt on interpretation of the constraints and objective. Some remodeling is therefore required before a problem of optimization is again achieved.

## 2. Some Traditional Approaches

Much can be said about how to address uncertainty in optimization, and how it should affect the modeling done in a specific application. But, the most fundamental idea is to begin by condensing random variables that depend on  $x$  back to numbers that depend on  $x$ . We will discuss several of the most familiar ways of doing this and compare their features. In the next section, a broader perspective will be taken and a theory furnishing guidelines will be developed.

In coming examples, we adopt the same approach in each case to the objective and every constraint, although approaches could be mixed in practice. This will underscore the principle of not thinking that constraints require different modes of treatment than objectives. It will also help to clarify shortcomings in these approaches.

### 2.1. Approach 1: Guessing the Future

A common approach in practice, serving essentially as a way of avoiding the issues, is to identify a single element  $\bar{\omega} \in \Omega$  as furnishing a best estimate of the unknown information, and then to

$$\text{minimize } c_0(x, \bar{\omega}) \quad \text{over all } x \in S \text{ satisfying } c_i(x, \bar{\omega}) \leq 0 \text{ for } i = 1, \dots, m. \quad (2.1)$$

Although this might be justifiable when the uncertainty is minor and well concentrated around  $\bar{\omega}$ , it is otherwise subject to serious criticism. A solution  $\bar{x}$  to (2.1) could lead, when the future state turns out to be some  $\omega$  other than  $\bar{\omega}$ , to a constraint value  $c_i(\bar{x}, \omega) > 0$ , or a cost  $c_0(\bar{x}, \omega)$  disagreeably higher than  $c_0(\bar{x}, \bar{\omega})$ . No provision has been made for the *risk* inherent in these eventualities. A decision  $\bar{x}$  coming out of (2.1) fails to hedge against the uncertainty and thus “puts all the eggs in one basket.” It does not incorporate any appraisal of how harmful an ultimate constraint violation or cost overrun might be to the application being modeled.

The weakness in this response to uncertainty can also be appreciated from another angle. If  $\Omega$  has been modeled as a continuum in a space  $\mathbb{R}^d$  of parameter vectors, the behavior of solutions to (2.1) as an optimization problem depending on  $\bar{\omega}$  as a parameter element could be poor. Even in linear programming it is well understood that tiny changes in coefficients can produce big changes, even jumps, in solutions. The dangers of not hedging could be seriously compounded by such instability.

### 2.2. Approach 2: Worst-Case Analysis

Another familiar approach is to rely on determining the worst that might happen. In its purest form, the counterpart to the deterministic problem (1.1) obtained in this way is to

$$\text{minimize } \sup_{\omega \in \Omega} c_0(x, \omega) \quad \text{over all } x \in S \text{ satisfying } \sup_{\omega \in \Omega} c_i(x, \omega) \leq 0 \text{ for } i = 1, \dots, m. \quad (2.2)$$

Here we write “sup” instead of “max” not only to avoid claiming attainment by some  $\omega$ , but also in deference to the technicality that we must be dealing with the essential least upper bound (neglecting sets of probability 0) when  $\Omega$  is infinite.

This very conservative formulation aims at ensuring that the constraints will be satisfied, no matter what the future brings. It devotes attention only to the worst possible outcomes, even if they are associated only with future states thought to be highly unlikely. Assessments of performance in more ordinary circumstances are not addressed.

Although the goal is to eliminate *all* risk, there is a price for that. The feasible set, consisting of the  $x$ s satisfying all the constraints, might be very small—possibly empty. For example, if the uncertainty in  $\omega$  has to do with storms, floods, or earthquakes, and  $x$  is tied to the design of a structure intended to withstand these forces, there may be no available choice of  $x$  guaranteeing absolute compliance. The optimizer may have to live with a balance between the practicality of  $x$  and the chance that the resulting design could be overpowered by some extreme event.



Nonetheless a strong attraction of this formulation is that the potential trouble over specifying a probability measure  $P$  on  $\Omega$  is effectively bypassed. A modern and more sophisticated version of the worst-case approach, motivated by that feature, is currently promoted as *robust optimization*. It aims likewise to avoid the introduction of a probability measure, but tries anyway to treat some parts of  $\Omega$  as more important (more likely) than others. A generalization along these lines will be given as Approach 8 in §6.4.

### 2.3. Approach 3: Relying on Expectations

Still another idea of long standing, back in the context of  $\Omega$  being a probability space, is to utilize the expectations of the random variables  $\underline{c}_i(x)$  as numbers that depend on  $x$ . Taking this approach at its purest, one could

$$\text{minimize } E[\underline{c}_0(x)] \quad \text{over all } x \in S \text{ satisfying } E[\underline{c}_i(x)] \leq 0 \text{ for } i = 1, \dots, m. \quad (2.3)$$

As far as the objective is concerned, this is a normal way of proceeding, and it has a long history. Yet for the constraints it seems generally ridiculous. If a constraint corresponded to the safety of a structure, for example, or the avoidance of bankruptcy, who would be satisfied with it only being fulfilled on the average? Expectations are primarily suitable for situations where the interest lies in long-range operation, and where stochastic ups and downs can safely average out. To the contrary, many applications have a distinctly short-run focus with serious risks in the foreground.

Why then should the expectation approach be acceptable for the objective in (2.3)? That runs counter to the no-distinction-in-treatment principle explained in the introduction. More will be seen about this below.

### 2.4. Approach 4: Standard Deviation Units as Safety Margins

An appealing way to dramatically improve on expectation constraints is to introduce safety margins based on standard deviation so as to ensure that the expected value is not just 0 but reassuringly below 0. For a choice of positive values  $\lambda_i > 0$ , the constraints set up in this manner take the form

$$\mu(\underline{c}_i(x)) + \lambda_i \sigma(\underline{c}_i(x)) \leq 0 \quad \text{for } i = 1, \dots, m. \quad (2.4)$$

The significance is that the future states  $\omega$  for which one gets  $c_i(x, \omega) > 0$  instead of the desired  $c_i(x, \omega) \leq 0$  correspond only to the upper part of the distribution of the random variable  $\underline{c}_i(x)$  that lies more than  $\lambda_i$  standard deviation units above the expected value of  $\underline{c}_i(x)$ . This resonates with many stipulations in statistical estimation about levels of confidence. Furthermore, it affords a compromise with the harsh conservatism of the worst-case approach.

What is the comparable formulation to (2.4) to adopt for the objective? The answer is to introduce another coefficient  $\lambda_0 > 0$  and

$$\text{minimize } \mu(\underline{c}_0(x)) + \lambda_0 \sigma(\underline{c}_0(x)) \quad \text{over all } x \in S \text{ satisfying (2.4).} \quad (2.5)$$

This is an interesting way to look at the objective, though it is almost never considered. Its interpretation, along the lines of (1.3) with the objective values viewed as costs is that one is looking for the lowest level of  $x_{n+1}$  as a cost threshold such that, for some  $x \in S$  satisfying (2.4), the cost outcomes  $c_0(x, \omega) > x_{n+1}$  will occur only in states  $\omega$  corresponding to the high end of the distribution of  $\underline{c}_i(x)$  lying more than  $\lambda_0$  standard deviation units above the mean cost.

Despite the seeming simplicity and attractiveness of this idea, it has a major flaw that will stand out when we get into the theoretical consideration of what guidelines should prevail for good modeling. A key property called coherency is lacking. A powerful substitute without this defect is presented in Approach 9 in §7.4.

## 2.5. Approach 5: Specifying Probabilities of Compliance

Another popular alternative to the worst-case approach, and which bears some resemblance to the one just outlined, is to pass to *probabilistic* constraints (also called *chance* constraints) in which the desired inequalities  $c_i(x, \omega) \leq 0$  are to hold at least with specified probabilities

$$\text{prob}\{\underline{c}_i(x) \leq 0\} \geq \alpha_i, \quad \text{for } i = 1, \dots, m, \quad (2.6)$$

where  $\alpha_i$  is a confidence level, say 0.99. Following the idea in (1.3) for handling the objective, the problem is to

$$\begin{aligned} &\text{minimize } x_{n+1} \text{ over all } (x, x_{n+1}) \in S \times \mathbb{R} \text{ satisfying} \\ &\quad \text{prob}\{\underline{c}_0(x) \leq x_{n+1}\} \geq \alpha_0 \text{ and the constraints (2.6).} \end{aligned} \quad (2.7)$$

For instance, with  $\alpha_0 = 0.5$  one would be choosing  $x$  to get the *median* of the random variable  $\underline{c}_0(x)$ , rather than its mean value, as low as possible.

Drawbacks are found even in this mode of optimization modeling, however. A qualitative objection, like the one about relying on a confidence level specified by standard deviation units, is that inadequate account is taken of the degree of danger inherent in potential violations beyond that level. In the cases where constraint violations  $c_i(x, \omega) > 0$  occur, which they do with probability  $1 - \alpha_i$ , is there merely inconvenience or a disaster? The specification of  $\alpha_i$ , alone, does not seem to fully address that. A technical objection too, from the optimization side, is that the probability expressions in (2.6) and (2.7) can exhibit poor mathematical behavior with respect to  $x$ , often lacking convexity and even continuity.

It is less apparent that Approach 5, like its predecessors, fits the pattern of condensing a random variable into a single number. Yet it does—in terms of quantiles and *value-at-risk*, a central idea in finance. In (2.6), the  $\alpha_i$ -quantile of the random variable  $\underline{c}_i(x)$  must be  $\leq 0$ , but technicalities can make the precise meaning problematic. This is discussed further in §5 when we explain value-at-risk and some recent approaches with *conditional value-at-risk* and its variants involving risk profiles.

## 2.6. Constraint Consolidation

Questions could be raised about the appropriateness of the tactic in Approaches 4 and 5 of putting a separate probability-dependent condition on each random variable. Why not, for instance, put  $\underline{c}_1(x), \dots, \underline{c}_m(x)$  into a single random variable

$$\underline{c}(x) \text{ with } c(x, \omega) = \max\{c_1(x, \omega), \dots, c_m(x, \omega)\} \quad (2.8)$$

and then constrain  $\text{prob}\{\underline{c}(x) \leq 0\} \geq \alpha$ ? That would correspond to insisting that  $x$  be feasible with probability at least  $\alpha$ . Nothing here should be regarded as counseling against that idea, which is very reasonable. However, the consequences may not be as simple as imagined.

The units in which the different costs  $c_i(x, \omega)$  are presented may be quite different. Issues of scaling could arise with implications for the behavior of a condition on  $\underline{c}(x)$  alone. Should each  $\underline{c}_i(x)$  be multiplied first by some  $\lambda_i > 0$  in (2.8) to adjust to this? If so, how should these coefficients be chosen?

Note also that individual constraints like those in (2.4) or (2.6) allow some costs to be subjected to tighter control than others, which is lost when they are consolidated into a single cost. A combination of individual constraints and a consolidated constraint may be appropriate. Not to be overlooked either is the no-distinction-in-treatment principle for the objective. But how should the objective be brought in?

Having raised these issues, we now put them in the background and continue with separate conditions on the random variables  $\underline{c}_i(x)$ . This theory will anyway be applicable to alternative formulations involving constraint consolidation.

## 2.7. Stochastic Programming and Multistage Futures

Stochastic programming is a major area of methodology dedicated to optimization problems under uncertainty (Wets [22]). Its leading virtue is extended modeling of the future, especially through *recourse decisions*. Simply put, instead of choosing  $x \in S$  and then having to cope with its consequences when a future state  $\omega \in \Omega$  is reached, there may be an opportunity then for a second decision  $x'$  that could counteract bad consequences, or take advantage of good consequences of  $x$ . There could then be passage to a state  $\omega'$  further in the future, and perhaps yet another decision  $x''$  after that. Although we will not go into this important subject, we note that the newer ideas explained here have yet to be incorporated in stochastic programming. When applied to our bare-bones format of choosing  $x$  and then experiencing  $\omega$ , the traditional formulation in stochastic programming would be to minimize the expectation of  $\underline{c}_0(x)$  over all  $x \in S$  satisfying  $\sup \underline{c}_i(x) \leq 0$  for  $i = 1, \dots, m$ . In particular, the objective and the constraints are treated quite differently. It should be clear from the comments about Approaches 2 and 3 that the improvements may need to be considered. Theoretical work in that direction has been initiated in Ruszczyński and Shapiro [21]. However, it should also be understood that stochastic programming strives, as far as possible in the modeling process, to eliminate uncertain constraints through the possibilities for recourse and various penalty expressions that might relate them. The purpose is not so much to obtain an exact solution as it is to identify ways of hedging that might otherwise be overlooked. The themes about risk which we develop here could assist further in that effort.

## 2.8. Dynamic Programming

Dynamic programming is another area of methodology in optimization under uncertainty that focuses on a future with many stages—perhaps an infinite number of stages. Dynamic programming operates backward in time to the present. Because it is more concerned with policies for controlling an uncertain system than coping with near-term risk, it is outside of the scope of this tutorial.

## 2.9. Penalty Staircases

A common and often effective approach to replacing the simple minimization of  $c_0(x)$  by something involving the random variable  $\underline{c}_0(x)$ , when uncertainty sets in, without merely passing to  $E[\underline{c}_0(x)]$ , is to

$$\text{minimize } E[\psi(\underline{c}_0(x))] \quad \text{for an increasing convex function } \psi \text{ on } (-\infty, \infty). \quad (2.9)$$

For example, a series of cost thresholds  $d_1, \dots, d_q$  might could be specified, and  $\psi$  could be taken to be a piecewise linear function having breakpoints at  $d_1, \dots, d_q$ , which imposes increasingly steeper penalization rates as successive thresholds are exceeded. A drawback is the difficulty in predicting how the selection of  $\psi$  will shape the distribution of  $\underline{c}_0(x)$  in optimality.

An alternative would be to proceed in the manner of Approach 5 with the choice of objective but to supplement it with constraints such as

$$\text{prob}\{\underline{c}_0^k(x) - d_k \leq 0\} \geq \alpha_0^k, \quad \text{for } k = 1, \dots, q. \quad (2.10)$$

The point is that a random variable like  $\underline{c}_0(x)$  can be propagated into a sequence of other random variables

$$\underline{c}_0^k(x) = \underline{c}_0(x) - d_k \quad \text{for } k = 1, \dots, q \quad (2.11)$$

in staircase fashion to achieve sharper control over results. Of course, the probabilistic constraints in (2.10) are only a temporary proposal, given the defects already discussed. Better ways of dealing with a *staircased random variable* as in (2.11) will soon be available.

### 3. Quantification of Risk

We wish to paint a large-scale picture of risk, without being bound to any one viewpoint or area of application, and to supply an axiomatic foundation that assists in clearing up some of the persistent misunderstandings.

What is risk? Everyone agrees that risk is associated with having to make a decision without fully knowing its consequences, due to future uncertainty, but also knowing that some of those consequences might be bad, or at least undesirable relative to others. Still, how might the *quantity* of risk be evaluated to construct a model in which optimization can be carried out?

Two basic ideas must be considered and coordinated. To many people, the amount of risk in a random variable representing a cost of some kind is the degree of *uncertainty* in it, i.e., how much it deviates from being constant. To other people, risk must be quantified in terms of a surrogate for the overall cost, such as its mean value, median value, or worst possible value. All the examples surveyed so far in optimization under uncertainty have revolved around such surrogates, but both ways of viewing risk will have roles in this tutorial.

The meaning of “cost” can be very general: Money, pollution, underperformance, safety hazard, failure to meet an obligation, etc. In optimization the concern is often a cost that is relative to some target and keeping it below 0, so that it does not become a “loss.” Of course, a negative cost or loss amounts to a “gain.”

For clarity, we will speak of *measures of deviation* when assessing inconstancy, with the standard deviation of a random variable serving as terminological inspiration. We will speak of *measures of risk* when assigning a single value to a random variable as a surrogate for its overall cost. Although this conforms to current common usage, it seems to create a competition between the second kind of measure and the first. It would really be more accurate to speak of the second kind as measures of the risk of *loss*, so we will use that terminology initially, before reverting to just speaking of measures of risk, for short.

Random variables could represent many things, but to achieve our goal of handling the random variables  $\underline{c}_i(x)$  in (1.2) that come out of an optimization problem such as (1.1) when uncertainty clouds the formulation, it is important to adopt an orientation. When speaking of a measure of risk of loss being applied to a random variable  $X$ , we will always have in mind that  $X$  represents a cost, as above: Positive outcomes  $X(\omega)$  of  $X$  are disliked, and large positive outcomes disliked even more, while negative outcomes are welcomed. This corresponds with traditions in optimization in which quantities are typically minimized or constrained to be  $\leq 0$ .

The core of the difficulty in optimization under uncertainty is the fact that a random variable is not, itself, a single quantity. The key to coping with this will be to condense the random variable into a single quantity by *quantifying the risk of loss*, rather than the degree of uncertainty, in it. We are thinking of positive outcomes of random variables  $\underline{c}_i(x)$  associated with constraints as losses (e.g., cost overruns). This presupposes that variables have been set up so that constraints are in  $\leq 0$  form. For  $\underline{c}_0(x)$ , associated with minimization, loss is unnecessary cost incurred by not making the best choices.

The random variables  $X$  in our framework are identified with functions from  $\Omega$  to  $\mathbb{R}$  that belong to the linear space  $\mathcal{L}^2$  which we introduced relative to a probability measure  $P$  on  $\Omega$ . They can be added, multiplied by scalars, and so forth. In quantifying loss, we must assign to each  $X \in \mathcal{L}^2$  a value  $\mathcal{R}(X)$ . We will take this value to belong to  $(-\infty, \infty]$ . In addition to having it be a real number, we may allow it in some circumstances to be  $\infty$ . Quantifying risk of loss will therefore revolve around specifying a functional  $\mathcal{R}$  from  $\mathcal{L}^2$  to  $(-\infty, \infty]$ . (In mathematics, a function on a space of functions, like  $\mathcal{L}^2$ , is typically called a functional.)

#### 3.1. A General Approach to Uncertainty in Optimization

In the context of problem (1.1) disrupted by uncertainty causing the function values  $c_i(x)$  to be replaced by the random variables  $\underline{c}_i(x)$  in (1.2), select for each  $i = 0, 1, \dots, m$  a functional

$\mathcal{R}_i: \mathcal{L}^2 \rightarrow (-\infty, \infty]$  aimed at quantifying the risk of loss. Then

$$\begin{aligned} &\text{replace the random variables } \underline{c}_i(x) \text{ by the functions } \bar{c}_i(x) = \mathcal{R}_i(\underline{c}_i(x)) \text{ and} \\ &\text{minimize } \bar{c}_0(x) \text{ over all } x \in S \text{ satisfying } \bar{c}_i(x) \leq 0 \text{ for } i = 1, \dots, m. \end{aligned} \quad (3.1)$$

As an important variant,  $\underline{c}_0(x)$  could be *staircased* in the manner of (2.11), and the same could be done for  $\underline{c}_1(x), \dots, \underline{c}_m(x)$ , if desired. This would introduce a multiplicity of random variables  $\underline{c}_i^k(x)$  that could individually be composed with functionals  $\mathcal{R}_i^k$  to supplement (3.1) with constraints providing additional control over the results of optimization.

Because the end product of any staircasing would still look like problem (3.1) except in notation, it will suffice, in the theory, to deal with (3.1).

The fundamental question now is this: What axiomatic properties should a functional have to be a good quantifier of the risk of loss? The pioneering work of Artzner et al. [3, 4], Delbaen [6], has provided a solid answer in terms of properties they identified as providing *coherency*. Their work concentrated on applications in finance, especially banking regulations, but the contribution goes far beyond that. We will present the concept in a form that follows more recent developments in expanding the ideas of those authors, or in some respects simplifying or filling in details. The differences will be discussed after the definition.

It will be convenient to use  $C$  to stand for either a number in  $\mathbb{R}$  or the corresponding constant random variable  $X \equiv C$  in  $\mathcal{L}^2$ . We write  $X \leq X'$  meaning that  $X(\omega) \leq X'(\omega)$  with probability 1, and so forth. The *magnitude* of an  $X \in \mathcal{L}^2$  is given by the norm

$$\|X\|_2 = (E[X^2])^{1/2} = (\mu^2(X) + \sigma^2(X))^{1/2}. \quad (3.2)$$

A sequence of random variables  $X^k$ ,  $k = 1, 2, \dots$ , converges to a random variable  $X$  with respect to this norm if  $\|X^k - X\|_2 \rightarrow 0$ , or equivalently, if both  $\mu(X^k - X) \rightarrow 0$  and  $\sigma(X^k - X) \rightarrow 0$  as  $k \rightarrow \infty$ .

### 3.2. Coherent Measures of Risk

A functional  $\mathcal{R}: \mathcal{L}^2 \rightarrow (-\infty, \infty]$  will be called a *coherent measure of risk in the extended sense* if

- (R1)  $\mathcal{R}(C) = C$  for all constants  $C$ ,
- (R2)  $\mathcal{R}((1 - \lambda)X + \lambda X') \leq (1 - \lambda)\mathcal{R}(X) + \lambda\mathcal{R}(X')$  for  $\lambda \in (0, 1)$  (“convexity”),
- (R3)  $\mathcal{R}(X) \leq \mathcal{R}(X')$  when  $X \leq X'$  (“monotonicity”),
- (R4)  $\mathcal{R}(X) \leq 0$  when  $\|X^k - X\|_2 \rightarrow 0$  with  $\mathcal{R}(X^k) \leq 0$  (“closedness”).

It will be called a *coherent measure of risk in the basic sense* if it also satisfies

- (R5)  $\mathcal{R}(\lambda X) = \lambda\mathcal{R}(X)$  for  $\lambda > 0$  (“positive homogeneity”).

The original definition of coherency in Artzner et al. [3, 4] required (R5). Insistence on this scaling property has since been called into question on various fronts. Although it is the dropping of it that we have in mind in supplementing their basic definition by an extended one, there are also other, lesser, differences between this version and the original definition.

Property (R1), which implies in particular that  $\mathcal{R}(0) = 0$ , has the motivation that if a random variable always has the same outcome  $C$ , then in condensing it to a single surrogate value, that value ought to be  $C$ . In Artzner et al. [3, 4] the place of (R1) was taken by a more complicated property, which was tailored to a banking concept, but came down to having

$$\mathcal{R}(X + C) = \mathcal{R}(X) + C \quad \text{for constants } C. \quad (3.3)$$

This extension of (R1) follows *automatically* from the combination of (R1) and (R2), as was shown in Rockafellar et al. [16]. In that paper, however, as well as in Artzner et al. [3, 4] the orientation was different: Random variables were viewed not as costs but as anticosts

(affording gains or rewards), which would amount to a switch of sign, transforming (3.3) into  $\mathcal{R}(X + C) = \mathcal{R}(X) - C$  and (R1) into  $\mathcal{R}(C) = -C$ . The formulation in this tutorial is dictated by the wish for a straightforward adaptation to the conventions of optimization theory, so that recent developments about risk can be at home in that subject and can smoothly reach a wider audience.

The risk inequality in (R3) has similarly been affected by the switch in orientation from anticosts to costs. The same will be true for a number of other conditions and formulas discussed or cited below, although this will not always be mentioned.

The combination of (R2) with (R5) leads to *subadditivity*

$$\mathcal{R}(X + X') \leq \mathcal{R}(X) + \mathcal{R}(X'). \quad (3.4)$$

On the other hand this property together with (R5) implies (R2). Subadditivity was emphasized in Artzner et al. [3, 4] as a key property that was lacking in the approaches popular with practioners in finance. The interpretation is that when  $X$  and  $X'$  are the loss variables (cost = loss) for two different portfolios, the total risk of loss should be reduced, or at least not made worse, when the portfolios are combined into one. This refers to *diversification*. The same argument can be offered as justification for the more basic property of convexity in (R2). Forming a weighted mixture of two portfolios should not increase overall loss potential. Otherwise there might be something to be gained by partitioning a portfolio (or comparable entity outside of finance) into increasingly smaller fractions.

The monotonicity in (R3) also makes perfect sense. If  $X(\omega) \leq X'(\omega)$  almost surely in the future states  $\omega$ , the risk of loss seen in  $X$  should not exceed the risk of loss seen in  $X'$ , with respect to quantification by  $\mathcal{R}$ . Yet some seemingly innocent strategies among practioners violate this, as will be seen shortly.

Note from applying (R3) in the case of  $X' = \sup X$ , when  $X$  is bounded from above, and invoking (R1), that

$$\mathcal{R}(X) \leq \sup X \quad \text{always,} \quad (3.5)$$

and, on the other hand, from taking  $X' = 0$  instead, that

$$\mathcal{R}(X) \leq 0 \quad \text{when } X \leq 0. \quad (3.6)$$

The latter property is in fact *equivalent* to the monotonicity in (R3) through the convexity in (R2).

### 3.3. Acceptable Risks

Artzner et al. [3, 4] introduced the terminology that the risk (of loss) associated with a random variable  $X$  is *acceptable* with respect to a choice of a coherent risk measure  $\mathcal{R}$  when  $\mathcal{R}(X) \leq 0$ . This ties in with the idea that  $\mathcal{R}(X)$  is the surrogate being adopted for the potential cost or loss. By (3.6),  $X$  is acceptable in particular if it exhibits no chance of producing a positive cost. But the concept allows compromise where there could sometimes be positive costs, as long as they are not overwhelming by some carefully chosen standard. The examples discussed in the next section explain this in greater detail.

In this respect, axiom (R4) says that if a random variable  $X$  can be approximated arbitrarily closely by acceptable random variables  $X^k$ , then  $X$  too should be acceptable. Such an approximation axiom was not included in the original papers of Artzner et al. [3, 4], but that may have been due to the fact that in those papers  $\Omega$  was a finite set and the finiteness of  $\mathcal{R}(X)$  was taken for granted. Because a finite convex function on a finite-dimensional space is automatically continuous, the closedness in (R5) would then be automatic. It has been noted by Ruszczyński that the continuity of  $\mathcal{R}$  follows also for infinite-dimensional  $\mathcal{L}^2$  from the combination of (R2) and (R3) as long as  $\mathcal{R}$  does not take on  $\infty$ . But  $\infty$  values can occur in some examples which we do not wish to exclude.

### 3.4. Coherency in Optimization

The main consequences of coherency in adopting (3.1) as a basic model for optimization under uncertainty are summarized as follows:

**Theorem 1.** *Suppose in problem (3.1), posed with functions  $\bar{c}_i(x) = \mathcal{R}_i(\underline{c}_i(x))$  for  $i = 0, 1, \dots, m$ , that each functional  $\mathcal{R}_i$  is a coherent measures of risk in the extended sense.*

(a) Preservation of convexity. *If  $c_i(x, \omega)$  is convex with respect to  $x$  for each  $\omega$ , then the function  $\bar{c}_i(x) = \mathcal{R}_i(\underline{c}_i(x))$  is convex. Thus, if problem (1.1) without uncertainty would have been a problem of convex programming, that advantage persists when uncertainty enters and is handled by passing to the formulation in (3.1) with coherency.*

(b) Preservation of certainty. *If  $\underline{c}_i(x)$  is actually just a constant random variable for each  $x$ , i.e.,  $c_i(x, \omega) = c_i(x)$  with no influence from  $\omega$ , then  $\bar{c}_i(x) = c_i(x)$ . Thus, the composition technique does not distort problem features that were not subject to uncertainty.*

(c) Insensitivity to scaling. *If the risk measures  $\mathcal{R}_i$  also satisfy (R5), then problem (3.1) remains the same when the units in which the values  $c_i(x, \omega)$  are denominated are rescaled.*

Property (a) of Theorem 1 holds through (R2) and (R3) because the composition of a convex function with a *nondecreasing* convex function is another convex function. (Without (R3), this could definitely fail.) Property (b) is immediate from (R1), whereas (c) corresponds to (R5).

Note that the constraints  $\bar{c}_i(x) \leq 0$  in problem (3.1) correspond to requiring  $x$  to *make the risk in the random variable  $\underline{c}_i(x)$  be acceptable* according to the dictates of the selected risk measure  $\mathcal{R}_i$ . A related feature coming out of (3.3), is that

$$\mathcal{R}_i(\underline{c}_i(x)) \leq b_i \iff \mathcal{R}_i(\underline{c}_i(x) - b_i) \leq 0. \quad (3.7)$$

Thus, acceptability is stable under translation and does not depend on where the zero is located in the scale of units for a random variable.

## 4. Coherency or Its Lack in Traditional Approaches

It is time to return to the traditional approaches to see how coherent they may or may not be. We will also look at the standards they implicitly adopt for deeming the risk in a cost random variable to be acceptable.

### 4.1. Approach 1: Guessing the Future

This corresponds to assessing the risk in  $\underline{c}_i(x)$  as  $\mathcal{R}(\underline{c}_i(x))$  with

$$\mathcal{R}(X) = X(\bar{\omega}) \text{ for some choice of } \bar{\omega} \in \Omega \text{ having positive probability.} \quad (4.1)$$

This functional  $\mathcal{R}$  does give a coherent measure of risk in the basic sense, but is open to criticism if used for such a purpose in responding to uncertainty. The risk in  $X$  is regarded as acceptable if there is no positive cost in the future state  $\bar{\omega}$ . No account is taken of any other future states.

### 4.2. Approach 2: Worst-Case Analysis

This corresponds to assessing the risk in  $\underline{c}_i(x)$  as  $\mathcal{R}(\underline{c}_i(x))$  with

$$\mathcal{R}(X) = \sup X. \quad (4.2)$$

Again, we have a coherent measure of risk in the basic sense, but it is severely conservative. Note that this furnishes an example where perhaps  $\mathcal{R}(X) = \infty$ . That will happen

whenever  $X$  does not have a finite upper bound (almost surely), which for finite  $\Omega$ , could not happen. The risk in  $X$  is acceptable only when  $X \leq 0$ , so that positive costs have zero probability.

### 4.3. Approach 3: Relying on Expectations

This corresponds to assessing the risk in  $\underline{c}_i(x)$  as  $\mathcal{R}(\underline{c}_i(x))$  with

$$\mathcal{R}(X) = \mu(X) = EX. \quad (4.3)$$

This is a coherent measure of risk in the basic sense, but it is feeble. Acceptability of the risk in  $X$  merely refers to negative costs being enough to balance out positive costs in the long run.

### 4.4. Approach 4: Standard Deviation Units as Safety Margins

This corresponds to assessing the risk in  $\underline{c}_i(x)$  as  $\mathcal{R}_i(\underline{c}_i(x))$  with

$$\mathcal{R}_i(X) = \mu(X) + \lambda_i \sigma(X) \quad \text{for some } \lambda_i > 0. \quad (4.4)$$

However, such a functional  $\mathcal{R}_i$  is *not* a coherent measure of risk. Axiom (R3) fails, although (R1), (R2), (R4) and even (R5) hold. This is one of the prime examples that the authors in Artzner et al. [3, 4] had in mind when developing their concept of coherency, because it lies at the heart of classical approaches to risk in finance.

Note that because (R3) fails for (4.4), the introduction of safety margins in this manner can *destroy convexity* when forming composites as in problem (3.1), and thus eliminate the benefits in part (a) of Theorem 1. This is unfortunate. Acceptability of the risk in  $X$  means that positive costs can only occur in the part of the distribution of  $X$  that lies more than  $\lambda_i$  standard deviation units above the mean. However, an excellent substitute that preserves convexity will emerge below in terms of conditional value-at-risk and other versions of safety margins based on various other measures of deviation.

### 4.5. Approach 5: Specifying Probabilities of Compliance

This corresponds to assessing the risk in  $\underline{c}_i(x)$  as  $\mathcal{R}_i(\underline{c}_i(x))$  with

$$\mathcal{R}_i(X) = q_{\alpha_i}(X) = \alpha_i\text{-quantile in the distribution of } X, \text{ for a choice of } \alpha_i \in (0, 1). \quad (4.5)$$

Although the precise meaning will be explained in the next section, it must be noted that this does *not* furnish a coherent measure of risk. The difficulty here lies in the convexity axiom (R2), which is equivalent to the combination of the positive homogeneity in (R5) and the subadditivity in (3.4). Although (R5) is obeyed, the subadditivity in (3.4), standing for the desirability of diversification, is violated. This was another important motivation for the development of coherency in Artzner et al. [3, 4]. Quantiles correspond in finance to value-at-risk, which is even incorporated into international banking regulations.

Without coherency, this approach, like the one before it, can *destroy convexity* that might otherwise be available for optimization modeling. Convexity can be salvaged in (4.5) if the distributions of the random variables  $\underline{c}_i(x)$  belong to the *log-concave* class for all  $x \in S$ , but even then there are technical hurdles because the convexity of  $\mathcal{R}_i$  is missing relative to the entire space  $\mathcal{L}^2$ .

For (4.5) acceptability of the risk in  $X$  means, of course, that positive costs are avoided with probability  $\alpha_i$ . Again, this is a natural idea. Although the faults in it are dismaying, *conditional* value-at-risk will address them.



## 5. Value-at-Risk and Conditional Value-at-Risk

In terms of the cumulative distribution function  $F_X$  of a random variable  $X$  and a probability level  $\alpha \in (0, 1)$ , the *value-at-risk*  $\text{VaR}_\alpha(X)$  and the  $\alpha$ -quantile  $q_\alpha(X)$  are identical:

$$\text{VaR}_\alpha(X) = q_\alpha(X) = \min\{z \mid F_X(z) \geq \alpha\}. \quad (5.1)$$

The *conditional value-at-risk*  $\text{CVaR}_\alpha(X)$  is defined by

$$\text{CVaR}_\alpha(X) = \text{expectation of } X \text{ in the conditional distribution of its upper } \alpha\text{-tail}, \quad (5.2)$$

so that, in particular,

$$\text{CVaR}_\alpha(X) \geq \text{VaR}_\alpha(X) \quad \text{always}. \quad (5.3)$$

The specification of what is meant by the “upper  $\alpha$ -tail” requires careful examination to clear up ambiguities. It should refer to the outcomes  $X(\omega)$  in the upper part of the range of  $X$  for which the probability is  $1 - \alpha$ . Ordinarily this would be the interval  $[q_\alpha(X), \infty)$ , but that is not possible when there is a probability atom of size  $\delta > 0$  at  $q_\alpha(X)$  itself (corresponding to  $F_X$  having a jump of such size at  $q_\alpha(X)$ ), because then  $\text{prob}[q_\alpha(X), \infty)$  is not necessarily  $1 - \alpha$  but rather something between  $\text{prob}(q_\alpha(X), \infty)$  and  $\text{prob}(q_\alpha(X), \infty) + \delta$ . The  $\alpha$ -tail conditional distribution cannot then be just the restriction of the  $P$  distribution to the interval  $[q_\alpha(X), \infty)$ , rescaled by dividing by  $1 - \alpha$ . Instead, that rescaling has to be applied to the distribution obtained on  $[q_\alpha(X), \infty)$  obtained by splitting the probability atom at  $q_\alpha(X)$  so as to leave just enough to bring the total probability up to  $1 - \alpha$ .

Although this is a foolproof definition for clarifying the concept, as introduced in Rockafellar and Uryasev [14] as a follow-up to the original definition of  $\text{CVaR}$  in Rockafellar and Uryasev [13], other formulas for  $\text{CVaR}_\alpha(X)$  may be operationally more convenient in some situations. One of these is

$$\text{CVaR}_\alpha(X) = \frac{1}{1 - \alpha} \int_\alpha^1 \text{VaR}_\beta(X) d\beta, \quad (5.4)$$

coming from Acerbi, cf. [1]. It has led to the term *average value-at-risk* being preferred for this concept in Föllmer and Schied [7], which has become a major reference in financial mathematics, although Acerbi preferred *expected shortfall* (a term with an orientation that conflicts with our cost view of random variables  $X$ ). The most potent formula for  $\text{CVaR}$  may be the *minimization rule*

$$\text{CVaR}_\alpha(X) = \min_{C \in \mathbb{R}} \{C + (1 - \alpha)^{-1} E[\max\{0, X - C\}]\}, \quad (5.5)$$

which was established in Rockafellar and Uryasev [13, 14]. It has the illuminating counterpart that

$$\text{VaR}_\alpha(X) = \text{left endpoint of } \arg \min_{C \in \mathbb{R}} \{C + (1 - \alpha)^{-1} E[\max\{0, X - C\}]\}. \quad (5.6)$$

In (5.5) and (5.6) a continuous convex function of  $C \in \mathbb{R}$  (dependent on  $X$  and  $\alpha$ ) is being minimized over  $\mathbb{R}$ . The  $\arg \min$  giving the values of  $C$  for which the minimum is attained is, in this special case, known always to be a nonempty, closed, bounded interval. Much of the time that interval collapses to a single value,  $\text{VaR}_\alpha(X)$ , but if not, then  $\text{VaR}_\alpha(X)$  is the lowest value in it.

We have posed in this formula in terms of  $\text{VaR}_\alpha(X)$  for harmony with  $\text{CVaR}_\alpha(X)$ , but it can easily be seen as an expression for calculating  $q_\alpha(X)$ . In that respect, it is equivalent to a formula of Koenker and Bassett [9], which is central to quantile regression (Koenker [8])

$$q_\alpha(X) = \text{left endpoint of } \arg \min_{C \in \mathbb{R}} E[\max\{0, X - C\} + (\alpha^{-1} - 1) \max\{0, C - X\}]. \quad (5.7)$$

Researchers in that area paid little attention to the minimum value in (5.5), but that value is primary here for the following reason.

**Theorem 2.** *For any probability level  $\alpha \in (0, 1)$ , the functional  $\mathcal{R}(X) = \text{CVaR}_\alpha(X)$  is a coherent measure of risk in the basic sense.*

This conclusion was reached from several directions. After the concept of conditional value-at-risk was introduced in Rockafellar and Uryasev [13] along with the minimization rule in (5.5), but initially only under the simplifying assumption that  $F_X$  had no jumps, Pflug [10] proved that a functional given by the right side of (5.5) would be a coherent measure of risk even if jumps were present. The fact that  $\text{CVaR}_\alpha(X)$ , if extended to the case of jumps by the careful interpretation of the  $\alpha$ -tail in (5.2), would still hold (5.5) was proved in Rockafellar and Uryasev [14]. Meanwhile, Acerbi and Tasche [2] showed the coherency of functionals expressed by the right side of (5.4), thus covering CVaR in another way.

Prior to these efforts, even with the strong case for coherent risk measures originally made in Artzner et al. [3, 4] essentially *no* good example of such a measure had been identified that had practical import beyond the theoretical. (In Artzner et al. [4], only general examples corresponding to the risk envelope formula to be presented in Theorem 4(a) were provided, without specifics. Other proposed measures such as tail risk, bearing a resemblance to conditional value-at-risk, but neglecting the complication of probability atoms, were, for a while, believed to offer coherence.) Currently, conditional value-at-risk, and other measures of risk constructed from it, form the core of the subject, especially in view of deep connections found in utility theory and second-order stochastic dominance.

A further property of conditional value-at-risk, distinguishing it from value-at-risk, is that for any single  $X \in \mathcal{L}^2$ ,

$$\begin{aligned} \text{CVaR}_\alpha(X) \text{ depends continuously on } \alpha \in (0, 1), \\ \text{with } \lim_{\alpha \rightarrow 1} \text{CVaR}_\alpha(X) = \sup X \text{ and } \lim_{\alpha \rightarrow 0} \text{CVaR}_\alpha(X) = EX. \end{aligned} \quad (5.8)$$

For value-at-risk, the limits are the same, but the dependence is not always continuous.

To appreciate the implications of conditional value-at-risk in approaching uncertainty, it will help to look first at what happens with value-at-risk itself. The crucial observation is that, through the definition in (5.1), one has

$$\text{prob}\{X \leq 0\} \geq \alpha \iff q_\alpha(X) \leq 0 \iff \text{VaR}_\alpha(X) \leq 0. \quad (5.9)$$

This can be used to rewrite the probabilistic constraints of Approach 5 in (2.6) and also the associated objective in (2.7), since

$$\text{prob}\{X \leq c\} \geq \alpha \iff q_\alpha \leq c \iff \text{VaR}_\alpha(X) \leq c. \quad (5.10)$$

In this way, Approach 5 can be expressed in the same kind of pattern as the others, where the random variables  $\underline{c}_i(x)$  are composed with some  $\mathcal{R}_i$  as in problem (3.1).

### 5.1. Approach 5, Recast: Safeguarding with Value-at-Risk

For a choice of probability levels  $\alpha_i \in (0, 1)$  for  $i = 0, 1, \dots, m$ ,

$$\begin{aligned} \text{minimize } \text{VaR}_{\alpha_0}(\underline{c}_0(x)) \quad \text{over all } x \in S \text{ satisfying} \\ \text{VaR}_{\alpha_i}(\underline{c}_i(x)) \leq 0 \text{ for } i = 1, \dots, m. \end{aligned} \quad (5.11)$$

In this case we have  $\mathcal{R}_i(X) = \text{VaR}_{\alpha_i}(X) = q_{\alpha_i}(X)$  and the original interpretations of this approach hold: We are asking in the constraints that the outcome of  $\underline{c}_i(x)$  should lie in  $(-\infty, 0]$  with probability at least  $\alpha_i$ , and subject to that, are seeking the lowest value  $c$  such that  $\underline{c}_0(x) \leq c$  with probability at least  $\alpha_0$ . But the shortcomings still hold as well:  $\mathcal{R}_i$  is *not* a coherent measure of risk, even though it satisfies (R1), (R3), (R4), and (R5). In looking to it for guidance, one could be advised paradoxically against diversification.

From a technical standpoint, other poor features of using  $\text{VaR}_\alpha(X)$ , or in equivalent notation  $q_\alpha(X)$ , to assess the overall risk in a random variable  $X$  are revealed. The formula in (5.1) predicts discontinuous behavior when dealing with random variables  $X$  whose distri-

bution functions  $F_X$  may have graphical flat spots or jumps, as is inevitable with discrete, and in particular, empirical distributions. All this is avoided by working with conditional value-at-risk instead. Those familiar with the fundamentals of optimization can immediately detect the root of the difference by comparing the formulas in (5.5) in (5.6). It is well known that the minimum value in an optimization problem dependent on parameters, even a problem in one dimension, as in this case, behaves much better than does the solution, or set of solution points. Thus,  $\text{CVaR}_\alpha(X)$  should behave better as a function of  $X$  than  $\text{VaR}_\alpha(X)$  and indeed it does.

## 5.2. Approach 6: Safeguarding with Conditional Value-at-Risk

For a choice of probability levels  $\alpha_i \in (0, 1)$  for  $i = 0, 1, \dots, m$ ,

$$\begin{aligned} &\text{minimize } \text{CVaR}_{\alpha_0}(\underline{c}_0(x)) \text{ over all } x \in S \text{ satisfying} \\ &\text{CVaR}_{\alpha_i}(\underline{c}_i(x)) \leq 0 \text{ for } i = 1, \dots, m. \end{aligned} \quad (5.12)$$

Here we use the coherent risk measures  $\mathcal{R}_i = \text{CVaR}_{\alpha_i}$ . What effect does this have on the interpretation of the model, in contrast to that of Approach 5, where  $\mathcal{R}_i = \text{VaR}_{\alpha_i}$ ? The conditional expectation in the definition of conditional value-at-risk provides the answer. However, due to the small complications that can arise over the meaning of the upper  $\alpha_i$ -tail of the random variable  $\underline{c}_i(x)$  when its distribution function may have a jump at the quantile value  $q_{\alpha_i}(\underline{c}_i(x)) = \text{VaR}_{\alpha_i}(\underline{c}_i(x))$ , it is best, for an initial understanding of the idea, to suppose there are no such jumps. Then,

$$\begin{aligned} \text{CVaR}_{\alpha_i}(\underline{c}_i(x)) \leq 0 \text{ means not merely that } \underline{c}_i(x) \leq 0 \\ \text{at least } 100\alpha_i\% \text{ of the time, but that the average of the} \\ \text{worst } 100(1 - \alpha_i)\% \text{ of all possible outcomes will be } \leq 0. \end{aligned} \quad (5.13)$$

Obviously, Approach 6 is, in this way, more cautious than Approach 5.

A valuable feature in Approach 6 is the availability of the minimization rule (5.5) for help in solving a problem in formulation (5.12). Insert this formula, with an additional optimization variable  $C_i$  for each index  $i$ , and the resultant problem to solve is to

$$\begin{aligned} &\text{find } (x, C_0, C_1, \dots, C_m) \in X \times \mathbb{R}^{m+1} \text{ to minimize} \\ &C_0 + (1 - \alpha_0)^{-1} E[\max\{0, \underline{c}_0(x) - C_0\}] \text{ subject to} \\ &C_i + (1 - \alpha_i)^{-1} E[\max\{0, \underline{c}_i(x) - C_i\}] \leq 0, \quad i = 1, \dots, m. \end{aligned} \quad (5.14)$$

Especially interesting is the case where each  $c_i(x, \omega)$  depends linearly (affinely) on  $x$ , and the space  $\Omega$  of future states  $\omega$  is finite. The expectations become weighted sums in which, through the introduction of still more variables, each max term can be replaced by a pair of linear inequalities so as to arrive at a *linear programming* reformulation of (5.14); cf. Rockafellar and Uryasev [14].

## 5.3. Staircasing

As a reminder, these approaches are being described in the direct picture of problem (3.1), but they also encompass the finer possibilities associated with breaking a random variable down into a staircased sequence, as in (2.11), obtained from a series of cost thresholds. (See comment after (3.1).)

## 6. Further Examples and Risk Envelope Duality

The examples of coherent measures of risk that we have accumulated so far are in Approaches 1, 2, 3, and 6. Other prime examples are provided in this section, including some that fit only the extended, not the basic, definition of coherency. Note, however, that any collection of examples automatically generates an even larger collection through the following operations, as is seen from the definition of coherency.

**Theorem 3.** *Coherency-preserving operations.*

(a) If  $\mathcal{R}_1, \dots, \mathcal{R}_r$  are coherent measures of risk in the basic sense, and if  $\lambda_1, \dots, \lambda_r$  are positive coefficients adding to 1, then a coherent measure of risk is defined by

$$\mathcal{R}(X) = \lambda_1 \mathcal{R}_1(X) + \lambda_2 \mathcal{R}_2(X) + \dots + \lambda_r \mathcal{R}_r(X). \quad (6.1)$$

Moreover, the same holds for coherent measures of risk in the extended sense.

(b) If  $\mathcal{R}_1, \dots, \mathcal{R}_r$  are coherent measures of of risk in the basic sense, then so too is

$$\mathcal{R}(X) = \max\{\mathcal{R}_1(X), \mathcal{R}_2(X), \dots, \mathcal{R}_r(X)\}. \quad (6.2)$$

Moreover, the same holds for coherent measures of risk in the extended sense.

### 6.1. Mixed CVaR and Spectral Profiles of Risk

An especially interesting example of such operations is *mixed conditional value-at-risk*, which refers to functionals having the form

$$\mathcal{R}(X) = \lambda_1 \text{CVaR}_{\alpha_1}(X) + \dots + \lambda_r \text{CVaR}_{\alpha_r}(X) \quad \text{with } \alpha_i \in (0, 1), \lambda_i > 0, \lambda_1 + \dots + \lambda_r = 1. \quad (6.3)$$

These functionals likewise furnish coherent measures of risk in the basic sense. One can even extend this formula to continuous sums

$$\mathcal{R}(X) = \int_0^1 \text{CVaR}_{\alpha}(X) d\lambda(\alpha). \quad (6.4)$$

This gives a coherent measure of risk in the basic sense for any weighting measure  $\lambda$  (with respect to generalized integration) that is nonnegative with total weight equal to 1. The formula in (6.3) corresponds to the discrete version of (6.4) in which a probability atom of size  $\lambda_i$  is placed at each  $\alpha_i$ . It has been proved (see Rockafellar et al. [16, Proposition 5]) that as long as  $\int_0^1 (1 - \alpha)^{-1} d\lambda(\alpha) < \infty$ , the measure of risk in (6.4) has an alternative expression in the form

$$\mathcal{R}(X) = \int_0^1 \text{VaR}_{\alpha}(X) \phi(\alpha) d\alpha \quad \text{with } \phi(\alpha) = \int_{(0, \alpha]} (1 - \beta)^{-1} d\lambda(\beta). \quad (6.5)$$

This is a *spectral representation*, in the sense of Acerbi [1], which relates to a dual theory of utility where  $\phi$  gives the *risk profile* of the decision maker.

Clearly, risk measures of form (6.3) can be used to approximate risk measures like (6.4). On the other hand, such risk measures can be supplied through the minimization rule (5.5) with a representation in terms of parameters  $C_1, \dots, C_r$  which is conducive to their practical use in optimization along the lines of the prescription at the end of the preceding section.

### 6.2. Approach 7: Safeguarding with Mixtures of Conditional Value-at-Risk

The use of single CVaR risk measures in Approach 6 could be expanded to mixtures as just described, with possible connections to risk profiles. All the properties in Theorem 1 would be available.

### 6.3. Risk Measures from Subdividing the Future

Let  $\Omega$  be partitioned into subsets  $\Omega_1, \dots, \Omega_r$  having positive probability, and for  $k = 1, \dots, r$  let

$$\mathcal{R}_k(X) = \sup_{\omega \in \Omega_k} X(\omega). \quad (6.6)$$

Then  $R_k$  is a coherent measure of risk in the basic sense (just like the one in Approach 2), and so too then, by Theorem 3(a), is

$$\mathcal{R}(X) = \lambda_1 \sup_{\omega \in \Omega_1} X(\omega) + \cdots + \lambda_r \sup_{\omega \in \Omega_r} X(\omega) \quad \text{for coefficients } \lambda_i > 0 \text{ adding to 1.} \quad (6.7)$$

The weights  $\lambda_k$  could be seen as lending different degrees of importance to different parts of  $\Omega$ . They could also be viewed as providing a sort of skeleton of probabilities to  $\Omega$ . A better understanding of this will be available below; see (6.16) and the surrounding explanation.

#### 6.4. Approach 8: Distributed Worst-Case Analysis

This refers to the modification of the worst-case formulation in Approach 2 to encompass risk measures of the forms in (6.6) and (6.7). Different partitions of  $\Omega$  might be used for different constraints.

#### 6.5. Risk Measures of Penalty Type

Another interesting way of quantifying the risk of loss is to modify the expected cost by adding a penalty term for positive costs. Recall that the so-called  $\mathcal{L}^p$ -norms are well defined as functionals on  $\mathcal{L}^2$  by

$$\|X\|_p = \begin{cases} E|X| & \text{for } p = 1, \\ (E[|X|^p])^{1/p} & \text{for } 1 < p < \infty, \\ \sup |X| & \text{for } p = \infty, \end{cases} \quad (6.8)$$

with  $\|X\|_p \leq \|X\|_{p'}$  when  $p \leq p'$ , but  $\|X\|_p$  can take on  $\infty$  when  $p > 2$  and  $\Omega$  is not finite (in which case  $\|\cdot\|_p$  is not technically a “norm” any more on  $\mathcal{L}^2$ ). Consider the functional  $\mathcal{R}: \mathcal{L}^2 \rightarrow (-\infty, \infty]$  defined by

$$\mathcal{R}(X) = EX + \lambda \|\max\{0, X - EX\}\|_p \quad \text{with } p \in [1, \infty], \lambda \in [0, 1]. \quad (6.9)$$

This too gives a coherent measure of risk in the basic sense. The coherency in (6.9) is not hard to verify up to a point: Axioms (R1), (R2), and (R4) are easily checked, along with (R5) for scalability. The monotonicity in (R3), however, is a bit more daunting. It is seen through the equivalence of (R3) with (3.6) using the inequality that

$$\|\max\{0, X - EX\}\|_p \leq \|\max\{0, X - EX\}\|_\infty = \sup X - EX$$

and the observation that  $EX + \lambda(\sup X - EX) \leq 0$  when  $X \leq 0$  and  $0 \leq \lambda \leq 1$ .

Often in financial applications where  $c_0(x, \omega)$  refers to the shortfall relative to a specified target level of profit, a penalty expression is like  $\|\max\{0, \underline{c}_0(x)\}\|_p$  is minimized, or such an expression raised to a power  $a > 1$ . This corresponds to composing  $\underline{c}_0(x)$  with  $\mathcal{R}(X) = \|\max\{0, X\}\|_p^a$ , which is *not* a coherent measure of risk. It satisfies (R2), (R3), (R4), and when  $a = 1$  even (R5). Only (R1) fails. The convexity preservation in Theorem 1(a) would hold, although not the certainty preservation in Theorem 1(b). A shortcoming is in the absence of a control over the expected value of  $\underline{c}_0(x)$ , which might even be positive. Minimum penalty might be achieved by a decision  $x$  in which there is a very high probability of loss, albeit not a big loss. By contrast, however, composition with a coherent risk measure such as in (6.9) would facilitate creating a safety margin against loss.

#### 6.6. Log-Exponential Risk Measures

Until now, every coherent measure of risk has satisfied (R5). Here is an important one that does not and therefore must be considered in the extended sense rather than the basic sense of coherency

$$\mathcal{R}(X) = \lambda \log E[e^{X/\lambda}] \quad \text{for a parameter value } \lambda > 0. \quad (6.10)$$

The confirmation of coherency in this case will be based on the duality theory presented next.

## 6.7. Representations of Risk Measures by Risk Envelopes

Coherent measures of risk can always be interpreted as coming from a kind of augmented worst-case analysis of expectations with respect to other probability measures  $P'$  on  $\Omega$  than the nominal one,  $P$ , or more specifically such measures  $P'$  having a well defined density  $Q = dP'/dP \in \mathcal{L}^2$  with respect to  $P$ . Such densities functions make up the set

$$\mathcal{P} = \{Q \in \mathcal{L}^2 \mid Q \geq 0, EQ = 1\}. \quad (6.11)$$

When  $Q$  is the density for  $P'$ , the expectation of a random variable  $X$  with respect to  $P'$  instead of  $P$  is  $E[XQ]$ , inasmuch as

$$E[XQ] = \int_{\Omega} X(\omega)Q(\omega) dP(\omega) = \int_{\Omega} X(\omega) \frac{dP'}{dP}(\omega) dP(\omega) = \int_{\Omega} X(\omega) dP'(\omega). \quad (6.12)$$

In this framework  $P$  itself corresponds to  $Q \equiv 1$ : We have  $EX = E[X \cdot 1]$ .

In contemplating a subset  $\mathcal{Q}$  of  $\mathcal{P}$ , one is essentially looking at some collection of alternatives  $P'$  to  $P$ . This could be motivated by a reluctance to accept  $P$  as furnishing a completely reliable model for the relative occurrences of the future states  $\omega \in \Omega$ , and the desire to test the dangers of too much trust in  $P$ .

A *risk envelope* will mean a nonempty, convex subset  $\mathcal{Q}$  of  $\mathcal{P}$  that is closed (so when elements  $Q^k$  of  $\mathcal{Q}$  converge to some  $Q$  in the  $\mathcal{L}^2$  sense described earlier,  $Q$  also belongs to  $\mathcal{Q}$ ). An *augmented risk envelope* will mean a risk envelope  $\mathcal{Q}$  supplied with a function  $a: \mathcal{Q} \rightarrow [0, \infty]$  (the *augmenting function*) having the properties that

$$\begin{cases} \text{the set } \{Q \in \mathcal{Q} \mid a(Q) < \infty\} \text{ has } \mathcal{Q} \text{ as its closure,} \\ \text{the set } \{Q \in \mathcal{Q} \mid a(Q) \leq C\} \text{ is closed for } C < \infty, \\ \text{the function } a \text{ is convex on } \mathcal{Q} \text{ with } \inf_{Q \in \mathcal{Q}} a(Q) = 0. \end{cases} \quad (6.13)$$

As a special case one could have  $a \equiv 0$  on  $\mathcal{Q}$ , and then the idea of an augmented risk envelope would simply reduce to that of a risk envelope by itself.

**Theorem 4.** *Dual characterization of coherency.*

(a)  $\mathcal{R}$  is a coherent measure of risk in the basic sense if and only if there is a risk envelope  $\mathcal{Q}$  (which will be uniquely determined) such that

$$\mathcal{R}(X) = \sup_{Q \in \mathcal{Q}} E[XQ]. \quad (6.14)$$

(b)  $\mathcal{R}$  is a coherent measure of risk in the extended sense if and only if there is a risk envelope  $\mathcal{Q}$  with an augmenting function  $a$  (both of which will be uniquely determined) such that

$$\mathcal{R}(X) = \sup_{Q \in \mathcal{Q}} \{E[XQ] - a(Q)\}. \quad (6.15)$$

The proof of this key result, reflecting a basic conjugacy principle in convex analysis (Rockafellar [11, 12]), can be found in a number of places, subject to variations on the underlying space (not always  $\mathcal{L}^2$ ). A version for the scalable case with  $\Omega$  finite appeared in Artzner et al. [4] and was elaborated for infinite  $\Omega$  in the unpublished exposition of Delbaen [6]. It was taken up specially for  $\mathcal{L}^2$  in Rockafellar et al. [17] where the term risk envelope was introduced. (The main results of that working paper were eventually published in Rockafellar et al. [16].) Versions without scalability are in Föllmer and Schied [7] and Rusczyński and Shapiro [20]. The condition in (6.13), that  $\inf_{\mathcal{Q}} a = 0$ , is essential for getting axiom (R1) to be satisfied in (6.15).

In view of the preceding discussion, formula (6.14) for the basic case has the interpretation that the risk  $\mathcal{R}(X)$  in  $X$  comes simply from a *worst-case analysis of the expected costs*  $E[XQ]$  corresponding to the probability measures  $P'$  having densities  $Q$  in the specified set  $\mathcal{Q}$ .

In short, selecting a coherent risk measure  $\mathcal{R}$  is equivalent to selecting a risk envelope  $\mathcal{Q}$ . Of course, this is rather black and white. Either a density  $Q$  presents a concern or it does not. The extended case with an augmenting function  $a$  provides a gradation in (6.15): Densities  $Q$  have less influence when  $a(Q) > 0$ .

Formula (6.14) would still give a coherent risk measure  $\mathcal{R}$  with  $\mathcal{Q}$  taken to be *any* nonempty subset  $\mathcal{Q}_0$  of  $\mathcal{P}$ , but that  $\mathcal{R}$  would also then be given by taking  $\mathcal{Q}$  to be the closed convex hull of  $\mathcal{Q}_0$  (the smallest closed convex subset of  $\mathcal{L}^2$  that includes  $\mathcal{Q}_0$ ). The assumption that  $\mathcal{Q}$  is already closed and convex makes it possible to claim that (6.12) furnishes a one-to-one correspondence  $\mathcal{R} \leftrightarrow \mathcal{Q}$ . A similar statement applies to formula (6.15).

## 6.8. Risk Envelope for Guessing the Future

The coherent (but hardly to be recommended) risk measure  $\mathcal{R}(X) = X(\bar{\omega})$  for a future state  $\bar{\omega}$  with  $\text{prob}(\bar{\omega}) > 0$  corresponds to taking  $\mathcal{Q}$  in (6.14) to consist of a single function  $Q$ , which has  $Q(\bar{\omega}) = 1/\text{prob}(\bar{\omega})$  but  $Q(\omega) = 0$  otherwise.

## 6.9. Risk Envelope for Worst-Case Analysis

The coherent risk measure  $\mathcal{R}(X) = \sup X$  corresponds to taking  $\mathcal{Q}$  in (6.14) to be all of  $\mathcal{P}$ , i.e., to consist of all  $Q \geq 0$  with  $EQ = 1$ .

## 6.10. Risk Envelope for Distributed Worst-Case Analysis

In the broader setting of  $\Omega$  being partitioned into subsets  $\Omega_k$  with weights  $\lambda_k$  as in (6.7), the risk envelope  $\mathcal{Q}$  consists of the densities  $Q$  with respect to  $P$  of the probability measures  $P'$  such that

$$P'(\Omega_k) = \lambda_k \quad \text{for } k = 1, \dots, r. \quad (6.16)$$

Not all probability measures alternative to  $P$  are admitted, as with ordinary worst-case analysis, but only those that conform to a specified framework of the likelihoods of different parts of  $\Omega$ . This provides a means for incorporating a rough structure of probabilities without having to go all the way to a particular measure like  $P$ , which serves here only in the technical background.

## 6.11. Risk Envelope for Relying on Expectations

The coherent risk measure for  $\mathcal{R}(X) = \mu(X)$  corresponds to taking  $\mathcal{Q}$  in (6.14) to consist solely of  $Q \equiv 1$ .

## 6.12. Risk Envelope for Standard Deviation Units as Safety Margins?

For the functional  $\mathcal{R}(X) = \mu(X) + \lambda\sigma(X)$  there is no risk envelope  $\mathcal{Q} \subset \mathcal{P}$ , due to the absence of coherency. However, because only (R3) fails, there is a representation in the form (6.14) involving elements  $Q$  that are not necessarily  $\geq 0$ . (See Rockafellar et al. [16].)

## 6.13. Risk Envelope for Safeguarding with Value-at-Risk, or in Other Words, for Specifying Probabilities of Compliance?

For  $\mathcal{R}(X) = q_\alpha(X) = \text{VaR}_\alpha(X)$  there is no corresponding risk envelope  $\mathcal{Q}$ , and in fact no representation in the pattern of (6.14), because  $\mathcal{R}$  lacks convexity.

## 6.14. Risk Envelope for Safeguarding with Conditional Value-at-Risk

For the functional  $\mathcal{R}(X) = \text{CVaR}_\alpha(X)$ , the risk envelope is

$$\mathcal{Q} = \{Q \in \mathcal{P} \mid Q \leq 1/\alpha\}. \quad (6.17)$$

This was first shown in Rockafellar et al. [17]. (See also Rockafellar et al. [16].)

### 6.15. Risk Envelope for Mixed Conditional Value-at-Risk

For  $\mathcal{R}(X) = \sum_{i=1}^r \lambda_i \text{CVaR}_{\alpha_i}(X)$  with positive weights  $\lambda_i$  adding to 1, the risk envelope is

$$\mathcal{Q} = \left\{ \sum_{i=1}^r \lambda_i Q_i \mid Q_i \in \mathcal{P}, 0 \leq Q_i \leq 1/\alpha_i \right\}. \quad (6.18)$$

Again, this comes from Rockafellar et al. [17]. (See also Rockafellar et al. [16].)

### 6.16. Risk Envelope for Measures of Penalty Type

For  $\mathcal{R}(X) = EX + \lambda \|\max\{0, X - EX\}\|_p$  with  $\lambda > 0$  and  $p \in [1, \infty]$ , the risk envelope is

$$\mathcal{Q} = \{Q \in \mathcal{P} \mid \|Q - \inf Q\|_q \leq 1\} \quad \text{where } q = \begin{cases} (1 - p^{-1})^{-1} & \text{when } p < \infty, \\ 1 & \text{when } p = \infty. \end{cases} \quad (6.19)$$

The proof of this is found in Rockafellar et al. [16, Examples 8 and 9]. (In all such references to the literature, the switch of orientation from  $X$  giving rewards to  $X$  giving costs requires a switch in signs.)

### 6.17. Augmented Risk Envelope for Log-Exponential Risk

The measure of risk expressed by  $\mathcal{R}(X) = \lambda \log E[e^{X/\lambda}]$ , which is not positively homogeneous, requires a risk envelope  $\mathcal{Q}$  with an augmenting function  $a$ , namely

$$\mathcal{Q} = \mathcal{P} \quad \text{with } a(Q) = \lambda E[Q \log Q]. \quad (6.20)$$

This recalls the duality between log-exponential functions and entropy-like functions which is well known in convex analysis, cf. Rockafellar [11]; Rockafellar and Wets [15, p. 482]. Its application to risk measures can be seen in Föllmer and Schied [7, p. 174], who refer to this  $\mathcal{R}$  as an “entropy” risk measure. The coherency of comes from showing that  $\mathcal{R}$  is obtained from the  $\mathcal{Q}$  and  $a$  in (6.20) by the formula in Theorem 4(b). In terms of the probability measure  $P'$  having density  $Q = dP'/dP$ , one has

$$E[Q \log Q] = I(P', P), \quad \text{the relative entropy of } P' \text{ over } P. \quad (6.21)$$

Ben-Tal and Teboulle [5] open this further and note that  $E[a(Q)]$  has been studied for more general  $a$  than  $a(Q) = [Q \log Q]$ , as in (6.20), for which they supply references.

## 7. Safety Margins and Measures of Deviation

Although the safety margins in Approach 4, using units of standard deviation, collide with coherency, the concept of a safety margin is too valuable to be ignored. The key idea behind it is to remedy the weakness of an expected cost constraint  $E[\underline{c}_i(x)] \leq 0$  by insisting on an adequate barrier between  $E[\underline{c}_i(x)]$  and 0. Is this ensured simply by passing to a constraint model  $\mathcal{R}_i(\underline{c}_i(x)) \leq 0$  for a coherent measure of risk  $\mathcal{R}_i$ , as we have been considering? Not necessarily. Guessing the future, with  $\mathcal{R}_i(X) = X(\bar{\omega})$  for some  $\bar{\omega}$  of positive probability, provides a quick counterexample. There is no reason to suppose that having  $X(\bar{\omega}) \leq 0$  entails having  $EX \leq 0$ . We have to impose some restriction on  $\mathcal{R}_i$  to get a safety margin. The following class of functionals must be brought in.



### 7.1. Averse Measures of Risk

Relative to the underlying probability measure  $P$  on  $\Omega$ , a functional  $\mathcal{R}: \mathcal{L}^2 \rightarrow (-\infty, \infty]$  will be called an *averse measure of risk in the extended sense*, if it satisfies axioms (R1), (R2), (R4) and

$$(R6) \quad \mathcal{R}(X) > EX \text{ for all nonconstant } X,$$

and *in the basic sense*, if it also satisfies (R5).

Recall that (R1) guarantees  $\mathcal{R}(X) = EX$  for constant  $X \equiv C$ . Aversity has the interpretation that *the risk of loss in a nonconstant random variable  $X$  cannot be acceptable unless, in particular,  $X(\omega) < 0$  on average*. Note that relations to expectation, and consequently to the particular choice of  $P$ , have not entered axiomatically until this point. (Averse measures of risk were initially introduced in Rockafellar et al. [17] in terms of “strict expectation-boundedness” rather than “aversity.” See also Rockafellar et al. [16].)

The monotonicity in (R3) has not been required in the definition, so an averse measure of risk might not be coherent. On the other hand, a coherent measure might not be averse, as the preceding illustration makes clear. In the end, we will want to focus on measures of risk that are simultaneously averse relative to  $P$  and coherent. At this stage, however, the concepts will come out clearer if we do not insist on that.

Averse measures of risk relative to  $P$  will be crucial in making the connection with the other fundamental way of quantifying the uncertainty in a random variable, namely its degree of deviation from constancy. Next, we develop this other kind of quantification axiomatically.

### 7.2. Measures of Deviation

A functional  $\mathcal{D}: \mathcal{L}^2 \rightarrow [0, \infty]$  will be called a *measure of deviation in the extended sense* if it satisfies

- (D1)  $\mathcal{D}(C) = 0$  for constants  $C$ , but  $\mathcal{D}(X) > 0$  for nonconstant  $X$ ,
- (D2)  $\mathcal{D}((1 - \lambda)X + \lambda X') \leq (1 - \lambda)\mathcal{D}(X) + \lambda\mathcal{D}(X')$  for  $\lambda \in (0, 1)$  (“convexity”).
- (D3)  $\mathcal{D}(X) \leq d$  when  $\|X^k - X\|_2 \rightarrow 0$  with  $\mathcal{D}(X^k) \leq d$  (“closedness”).

It will be called a *measure of deviation in the basic sense* when furthermore

$$(D4) \quad \mathcal{D}(\lambda X) = \lambda\mathcal{D}(X) \text{ for } \lambda > 0 \text{ (“positive homogeneity”).}$$

Either way, it will be called *coherent* if it also satisfies

$$(D5) \quad \mathcal{D}(X) \leq \sup X - EX \text{ for all } X \text{ (“upper range boundedness”).}$$

An immediate example of a measure of deviation in the basic sense is standard deviation,  $\mathcal{D}(X) = \sigma(X)$ . In addition to (D1), (D2), and (D3), it satisfies (D4), but *not*, as it turns out, (D5). The definition aims to quantify the uncertainty, or nonconstancy, of  $X$  in different ways than just standard deviation, allowing even for cases where  $\mathcal{D}(X)$  might not be the same as  $\mathcal{D}(-X)$ . The reason for tying (D5) to  $\mathcal{D}$  being “coherent” is revealed by the theorem below.

### 7.3. Risk Measures Versus Deviation Measures

**Theorem 5.** *A one-to-one correspondence between measures of deviation  $\mathcal{D}$  in the extended sense and averse measures of risk  $\mathcal{R}$  in the extended sense is expressed by the relations*

$$\mathcal{R}(X) = EX + \mathcal{D}(X), \quad \mathcal{D}(X) = \mathcal{R}(X - EX), \quad (7.1)$$

*with respect to which*

$$\mathcal{R} \text{ is coherent} \iff \mathcal{D} \text{ is coherent.} \quad (7.2)$$

*In this correspondence, measures in the basic sense are preserved:*

$$\mathcal{R} \text{ is positively homogeneous} \iff \mathcal{D} \text{ is positively homogeneous.} \quad (7.3)$$

This result, for the basic case, was originally obtained in the working paper Rockafellar et al. [17] and finally in Rockafellar et al. [16]. Extension of the proof beyond positive homogeneity is elementary. The risk envelope representation of Theorem 4 for  $\mathcal{R}$  under coherency immediately translates, when  $\mathcal{R}$  is strict, to a similar representation for the associated  $\mathcal{D}$

$$\mathcal{D}(X) = \sup_{Q \in \mathcal{Q}} E[(X - EX)Q] \quad \text{in the basic case} \quad (7.4)$$

or, on the other hand

$$\mathcal{D}(X) = \sup_{Q \in \mathcal{Q}} \{E[(X - EX)Q] - E[a(Q)]\} \quad \text{in the extended case.} \quad (7.5)$$

It follows from Theorem 5 that a deviation measure  $\mathcal{D}$  is coherent *if and only if* it has a representation of this kind (necessarily unique) in which  $\mathcal{Q}$  and  $a$  (identically 0 in the basic case) meet the specifications in (6.13). Deviation measures that are not coherent have representations along the same lines, but with  $\mathcal{Q} \not\subset \mathcal{P}$ . The elements  $Q \in \mathcal{Q}$  still have  $EQ = 1$ , but  $Q \not\geq 0$  for some, and their interpretation in terms of densities  $dP'/dP$  of probability measures  $P'$  being compared to  $P$  drops away. For more on this, see Rockafellar et al. [16].

The fact that standard deviation  $\mathcal{D}(X) = \sigma(X)$  does not have a risk envelope representation (7.4) with  $\mathcal{Q} \subset \mathcal{P}$  lies behind the assertion that this deviation measure is not coherent and, at the same time, confirms the lack of that property in Approach 4. This shortcoming of  $\mathcal{D}(X) = \sigma(X)$  can also be gleaned from the condition for  $\mathcal{D}$  to be coherent in (D5). If  $\sigma(X) \leq \sup X - EX$  for all  $X$ , we would also have by applying this to  $-X$ , that  $\sigma(X) \leq EX - \inf X$ , and therefore  $\sigma(X) \leq [\sup X - \inf X]/2$  for all random variables  $X$ , which is in general false.

Theorem 5 shows that to introduce safety margins in optimization under uncertainty without falling into the trap of Approach 4 with its lack of coherency, we must pass from standard deviation units to those of some other measure of deviation satisfying (D5). Here we can take advantage of the fact that when  $\mathcal{D}$  is a coherent measure of deviation, then so too is  $\lambda\mathcal{D}$  for any  $\lambda > 0$ .

#### 7.4. Approach 9: Generalized Deviation Units as Safety Margins

Faced with the random variables (1.2), choose deviation measures  $\mathcal{D}_i$  for  $i = 0, 1, \dots, m$  with coefficients  $\lambda_i > 0$ . Pose the constraints in the form

$$\mu(\underline{c}_i(x)) + \lambda_i \mathcal{D}_i(\underline{c}_i(x)) \leq 0 \quad \text{for } i = 1, \dots, m, \quad (7.6)$$

thus requiring that positive outcomes of  $\underline{c}_i(x)$  can only occur in the part of the range of this random variable lying more than  $\lambda_i$  deviation units, with respect to  $\mathcal{D}_i$ , above the mean  $\mu(\underline{c}_i(x)) = E[\underline{c}_i(x)]$ . The goal is to

$$\begin{aligned} &\text{minimize } \mu(\underline{c}_0(x) - x_{n+1}) + \lambda_0 \mathcal{D}_0(\underline{c}_0(x) - x_{n+1}) \\ &\text{over all } (x, x_{n+1}) \in S \times \mathbb{R} \text{ satisfying (7.4).} \end{aligned} \quad (7.7)$$

This mimics Approach 4 with  $\sigma(X)$  replaced by  $\mathcal{D}_i(X)$ . The measures of risk filling the role prescribed in (3.1) now have the form

$$\mathcal{R}_i(X) = \mu(X) + \lambda_i \mathcal{D}_i(X) = EX + \mathcal{D}'_i(X) \quad \text{for } i = 1, \dots, m, \text{ with } \mathcal{D}'_i = \lambda_i \mathcal{D}_i. \quad (7.8)$$

The contrast between this and the previous case, where  $\mathcal{D}_i(X) = \sigma(X)$ , is that  $\mathcal{R}_i$  is coherent when  $\mathcal{D}_i$  is coherent. Hence, if this holds for  $i = 0, 1, \dots, m$ , the optimization properties in Theorem 1 apply to problem (7.7).

Theorem 5 provides the right that all the approaches to optimization under uncertainty considered so far in the mode of (3.1) in which the  $\mathcal{R}_i$ s are *averse* measures of risk correspond

to introducing safety margins in units of generalized deviation. But this is not true when the  $\mathcal{R}_i$ s are not averse. Because we want the  $\mathcal{R}_i$ s to be coherent as well, the question arises: What examples do we have at this point of *coherent* measures of risk that are also *averse* (relative to  $P$ )? Those obtained via (7.8) from a coherent measure of deviation serve the purpose, but the issue then devolves to looking for examples of such measures of deviation.

### 7.5. Aversity of CVaR and Mixed CVaR

The coherent risk measure  $\mathcal{R}(X) = \text{CVaR}_\alpha(X)$  is averse for any  $\alpha \in (0, 1)$ . The corresponding coherent measure of deviation is

$$\mathcal{D}(X) = \text{CVaR}_\alpha(X - EX). \quad (7.9)$$

More generally, any mixture  $\mathcal{R}(X) = \lambda_1 \text{CVaR}_{\alpha_1}(X) + \cdots + \lambda_r \text{CVaR}_{\alpha_r}(X)$  with positive weights adding to 1 gives an averse, coherent measure partnered with the deviation measure

$$\mathcal{D}(X) = \lambda_1 \text{CVaR}_{\alpha_1}(X - EX) + \cdots + \lambda_r \text{CVaR}_{\alpha_r}(X - EX), \quad (7.10)$$

which therefore is coherent also.

### 7.6. Aversity of Risk Measures of Penalty Type

The coherent risk measure  $\mathcal{R}(X) = EX + \lambda \|\max\{0, X - EX\}\|_p$  for any  $\lambda > 0$  and any  $p \in [1, \infty]$  is averse. That is clear from the definition of strictness through (R6): We do have  $\mathcal{R}(X) - EX > 0$  unless  $X$  is constant. The corresponding coherent measure of deviation is

$$\mathcal{D}(X) = \lambda \|\max\{0, X - EX\}\|_p. \quad (7.11)$$

### 7.7. Aversity of the Worst-Case Risk Measure

The coherent risk measure  $\mathcal{R}(X) = \sup X$  is averse, again directly through the observation that it satisfies (R6): Except when  $X$  is constant, we always have  $\sup X > EX$ . This can also be viewed as a special case of the previous example because  $\|\max\{0, X - EX\}\|_\infty = \sup X - EX$ . We see then as well that the corresponding coherent measure of deviation is

$$\mathcal{D}(X) = \sup X - EX. \quad (7.12)$$

### 7.8. Aversity of Distributed Worst-Case Risk Measures

The coherent risk measure in (6.7) is averse, with

$$\mathcal{D}(X) = -EX + \lambda_1 \sup_{\omega \in \Omega_1} X(\omega) + \cdots + \lambda_r \sup_{\omega \in \Omega_r} X(\omega). \quad (7.13)$$

### 7.9. Aversity of Log-Exponential Risk Measures

The coherent risk measure in the extended sense given by  $\mathcal{R}(X) = \lambda \log E[e^{X/\lambda}]$  is averse. Direct verification through (R6) works again: Since  $EX = \lambda \log E[e^{EX/\lambda}]$ , having  $\mathcal{R}(X) > EX$  amounts to having  $E[e^Y] > e^{EY}$  for  $Y = X/\lambda$ , and that is Jensen's inequality for a nonconstant  $Y$  and the strictly convex function  $t \mapsto e^t$ . We conclude that a coherent measure of deviation in the extended sense is furnished by

$$\mathcal{D}(X) = \lambda \log E[e^{(X-EX)/\lambda}] \quad \text{for any } \lambda > 0. \quad (7.14)$$

### 7.10. Another Example of a Deviation Measure in the Extended Sense

A deviation measure of a type related to so-called robust statistics is defined in terms of a parameter  $s > 0$  by

$$\mathcal{D}(X) = \begin{cases} \sigma^2(X) & \text{if } \sigma(X) \leq s, \\ s^2 + 2s[\sigma(X) - s] & \text{if } \sigma(X) \geq s. \end{cases}$$

Here  $\sigma(X)$  could be replaced by  $\mathcal{D}_0(X)$  for any deviation measure  $\mathcal{D}_0$ .

## 8. Characterizations of Optimality

For a problem of optimization in the form of (3.1) with each  $\mathcal{R}_i$  a coherent measure of risk, how can solutions  $\bar{x}$  be characterized? This topic could lead to a major discussion, but here we only have space for a few words. Basically this requires working with subgradients of the functions  $\bar{c}_i$  in the sense of convex analysis, and that means being able to determine the subgradients of the functionals  $\mathcal{R}_i$ . That has been done in Rockafellar et al. [18]. The answer, for the case of  $\mathcal{R}_i$  being positively homogeneous, is given in terms of the corresponding risk envelopes  $\mathcal{Q}_i$ . The set of subgradients  $Y$  of  $\mathcal{R}_i$  at  $X$  is

$$\partial \mathcal{R}_i(X) = \arg \max_{Q \in \mathcal{Q}_i} E[XQ]. \quad (8.1)$$

Details of what that means for various examples are provided in Rockafellar et al. [18]. Fundamentally, Lagrange multipliers, duality, and other important ideas in convex optimization revolve around the risk envelopes when invoked in the context of uncertainty.

Note, for instance, that if the random variable  $\underline{c}_0(x)$  is staircased as in (2.11) and constraints of the form  $\text{CVaR}_{\alpha_k}(\underline{c}_0(x) - d_k) \leq 0$  are imposed to tune its distribution, a Lagrangian expression in the form

$$\text{CVaR}_{\alpha_0}(\underline{c}_0(x)) + y_1[\text{CVaR}_{\alpha_1}(\underline{c}_0(x)) - d_1] + \cdots + y_q[\text{CVaR}_{\alpha_q}(\underline{c}_0(x)) - d_q] \quad (8.2)$$

is generated in which minimization in  $x$  for fixed nonnegative multipliers  $y_1, \dots, y_q$  corresponds to minimization of  $\mathcal{R}(\underline{c}_0(x))$  for the mixed CVaR risk measure

$$\mathcal{R} = \lambda_0 \text{CVaR}_{\alpha_0} + \lambda_1 \text{CVaR}_{\alpha_1} + \cdots + \lambda_q \text{CVaR}_{\alpha_q} \quad (8.3)$$

in which the coefficient vector  $(\lambda_0, \lambda_1, \dots, \lambda_q)$  is obtained by rescaling  $(1, y_1, \dots, y_q)$  so that the coordinates add to 1. Duality, in the framework of identifying the multipliers which yield optimality, must in effect identify the weights in this mixture and therefore an implicit risk profile for the optimizer who imposed the staircase constraints.

## Acknowledgments

This research was partially supported by National Science Foundation grant DMI 0457473, Percentile-Based Risk Management Approaches in Discrete Decision Making Problems.

## References

- [1] C. Acerbi. Spectral measures of risk: A coherent representation of subjective risk aversion. *Journal of Banking and Finance* 26:1505–1518, 2002.
- [2] C. Acerbi and D. Tasche. On the coherence of expected shortfall. *Journal of Banking and Finance* 26:1487–1503, 2002.
- [3] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath. Thinking coherently. *Risk* 10:68–91, 1997.
- [4] P. Artzner, F. Delbaen, J.-M. Eber, and D. Heath. Coherent measures of risk. *Mathematical Finance* 9:203–227, 1999.
- [5] A. Ben Tal and M. Teboulle. An old-new concept of convex risk measures: The optimized certainty equivalent. *Mathematical Finance*. 17:449–476, 2007.

- [6] F. Delbaen. Coherent risk measures on general probability spaces. Working paper, Eidgenössische Technische Hochschule, Zürich, Switzerland, <http://www.math.ethz.ch/~delbaen/ftp/preprints/RiskMeasuresGeneralSpaces.pdf>, 2000.
- [7] H. Föllmer and A. Schied. *Stochastic Finance*. Walter de Gruyter, Berlin, Germany, 2002.
- [8] R. Koenker. *Quantile Regression*. Econometric Society Monograph Series, Cambridge University Press, West Nyack, NY, 2005.
- [9] R. Koenker and G. W. Bassett. Regression quantiles. *Econometrica* 46:33–50, 1978.
- [10] G. Pflug. Some remarks on the value-at-risk and the conditional value-at-risk. S. Uryasev, ed. *Probabilistic Constrained Optimization: Methodology and Applications*. Kluwer Academic Publishers, Norwell, MA, 2000.
- [11] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, NJ, 1970.
- [12] R. T. Rockafellar. *Conjugate Duality and Optimization*, No. 16 in *Conference Board of Math. Sciences Series*. SIAM, Philadelphia, PA, 1974.
- [13] R. T. Rockafellar and S. P. Uryasev. Optimization of conditional value-at-risk. *Journal of Risk* 2:21–42, 2000.
- [14] R. T. Rockafellar and S. P. Uryasev. Conditional value-at-risk for general loss distributions. *Journal of Banking and Finance* 26:1443–1471, 2002.
- [15] R. T. Rockafellar and R. J-B Wets. *Variational Analysis*, No. 317 in the series *Grundlehren der Mathematischen Wissenschaften*. Springer-Verlag, Berlin, Germany, 1997.
- [16] R. T. Rockafellar, S. Uryasev, and M. Zabarankin. Generalized deviations in risk analysis. *Finance and Stochastics* 10:51–74, 2006.
- [17] R. T. Rockafellar, S. Uryasev, and M. Zabarankin. Deviation measures in risk analysis and optimization. Research Report 2002–7, Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL, 2002.
- [18] R. T. Rockafellar, S. Uryasev, and M. Zabarankin. Optimality conditions in portfolio analysis with general deviation measures. *Mathematical Programming, Series B* 108:515–540, 2006.
- [19] R. T. Rockafellar, S. Uryasev, and M. Zabarankin. Master funds in portfolio analysis with general deviation measures. *Journal of Banking and Finance* 30:743–778, 2006.
- [20] A. Ruszczyński and A. Shapiro. Optimization of convex risk functions. *Mathematics of Operations Research* 31:433–452, 2006.
- [21] A. Ruszczyński and A. Shapiro. Conditional risk mappings. *Mathematics of Operations Research* 31:544–561, 2006.
- [22] R. J-B Wets. Stochastic programming. G. Nemhauser and A. Rinnooy Kan, eds. *Handbook of Operations Research and Management Science*, Vol. 1, *Optimization*. Elsevier Science Publishers, Amsterdam, The Netherlands, 573–629, 1987.
- [23] A. A. Trindade, S. Uryasev, A. Shapiro, and G. Zrazhevsky. Financial prediction with constrained tail risk. *Journal of Banking and Finance*. Forthcoming.

# Differential Games in Marketing Science

*Gary M. Erickson*

Marketing and International Business Department, University of Washington Business School,  
Seattle, Washington 98195, erick@u.washington.edu

**Abstract** Markets are dynamic and involve oligopolistic competition, so the fit between differential games and marketing science is a natural one. This tutorial presents an overview of differential games, and discusses examples of differential-game models used in marketing science, focusing on models involving advertising. The challenge of applying differential games empirically is also discussed, and an example application is presented.

**Keywords** differential games; marketing; empirical application

---

## 1. Introduction

Over the past thirty years, the methodology of differential games has become an important modeling framework in the marketing science literature. Differential-game formulations of marketing problems are useful because of two key aspects of markets and the management of marketing decision variables: Markets are constantly changing, and markets involve oligopolistic competition. Differential games, as a fusion of game theory and dynamic optimization, allow marketers to view their decisions and outcomes in the double-featured context of competition and change.

The purpose of this tutorial on differential games in marketing science is primarily to introduce the reader to essential basics of differential games, and to discuss various differential game models that have been employed in marketing. Further, the empirical applicability of differential games is illustrated. The aim is not to provide a comprehensive review of differential games in marketing settings; such can be found in Jørgensen and Zaccour [11], Erickson [7], and Dockner et al. [4]. The remainder of the tutorial proceeds as follows. The next section outlines the essential basics of differential games and provides an example. Section 3 discusses four differential game models that involve advertising. Section 4 provides empirical applications, and the final section offers conclusions.

## 2. Differential Game Essentials

A differential game is a game played in continuous time, and involves two key types of variables, control variables, and state variables. In marketing, control variables are advertising expenditures and/or prices of competing products, and state variables are outcomes such as sales or market share. Both types of variables are considered to be functions of time, so that the decision problem for each player is to establish a continuous path of control variable values across a time horizon, and state variables evolve continuously over the horizon, the evolution being influenced by the control variables.

Denote time by  $t$  and consider that we have  $n$  players,  $n$  being an integer greater than or equal to 2. Each player  $i \in \{1, \dots, n\}$  has a control variable  $u_i(t)$ , which in general is a vector, and is subject to the constraint  $u_i(t) \in U_i(t, x(t))$ . The state of the system at time  $t$  is characterized as an  $m$ -vector  $x(t) = (x_1(t), \dots, x_m(t))'$ . Initial values  $x(0)$  at  $t = 0$  are

assumed given and fixed. The state of the system evolves through a system of differential equations:

$$\dot{x}(t) = \frac{dx(t)}{dt} = f(t, x(t), u_1(t), \dots, u_n(t)), \quad x(0) = x_0. \quad (1)$$

The rate of change of the state variables depends on time, the state of the system, and the controls of the  $n$  players.

At time  $t$ , the payoff to each player  $i$  depends on  $t$ , the state of the system, and the controls:

$$g_i(t, x(t), u_1(t), \dots, u_n(t)). \quad (2)$$

The objective of each player is to maximize the present value of its payoff stream

$$J_i(u_1(\cdot), \dots, u_n(\cdot)) = \int_0^T e^{-r_i t} g_i(t, x(t), u_1(t), \dots, u_n(t)) dt + e^{-r_i T} S_i(x(T)), \quad (3)$$

where  $r_i$  is the discount rate of player  $i$ , and  $S_i(x(T))$  is a salvage value for  $i$ . The time horizon  $T$  can be infinite,  $T = \infty$ , in which case a salvage value is not included or needed.

## 2.1. Nash Equilibrium

Call  $\phi_i$  the strategy player  $i$  uses to obtain chosen  $u_i(t)$  values. The strategy  $\phi_i$  may depend only on time,  $\phi_i(t)$ , in which case the strategy is termed an *open-loop* strategy, or it may depend on the current state of the system as well,  $\phi_i(t, x(t))$ , which is termed a *feedback* (also *closed-loop* as in Erickson [7] and *Markovian* as in Jørgensen and Zaccour [11]) strategy. Further, if we have an infinite horizon, and the functions  $f$  and  $g_i$  in (1) and (2) do not depend explicitly on time  $t$ , and so have an *autonomous* differential game, the feedback strategy may appropriately be considered a function only of the state of the system,  $\phi_i(x(t))$ , a *stationary* feedback strategy.

The players are assumed to choose their strategies simultaneously, cannot come to binding agreements, have complete information about all the objective functions, and each player is aware that the other players are also choosing control values to maximize their respective situations. A *Nash equilibrium* results when each player maximizes its payoff given what it determines to be the other players' chosen strategies. Formally, a set of strategies  $(\phi_1, \dots, \phi_n)$  constitutes a Nash equilibrium if

$$J_i(\phi_1, \dots, \phi_n) \geq J_i(\phi_1, \dots, \phi_{i-1}, u_i, \phi_{i+1}, \dots, \phi_n), \quad \forall u_i \in U_i, \quad \forall i \in \{1, \dots, n\}. \quad (4)$$

## 2.2. Determination of Nash Equilibria

There are two approaches one can use to develop Nash equilibrium strategies. One involves optimal control methods, with Hamiltonians and costate variables, and the other follows dynamic programming principles, with value functions and Hamilton-Jacobi-Bellman equations. Optimal control is used to determine open-loop strategies, and dynamic programming is a better approach for feedback strategies.

The optimal control approach calls for a Hamiltonian for each player

$$H_i(t, x(t), u_1(t), \dots, u_n(t), \lambda_i(t)) = g_i + \lambda_i' f, \quad (5)$$

where  $\lambda_i$  is a vector of  $m$  costate variables. For strategies that are a function of time only, the Nash equilibrium conditions (4) lead, through application of optimal control, to the following necessary conditions:

$$\frac{\partial H_i}{\partial u_i} = 0, \quad \dot{\lambda}_i = r_i \lambda_i - \frac{\partial H_i}{\partial x}, \quad \lambda_i(T) = \frac{\partial S_i}{\partial x}(x(T)), \quad i = 1, \dots, n. \quad (6)$$

The conditions (6), along with the state dynamics (1), form a two-point boundary value problem (TPBVP), which is solvable through numerical methods. If the time horizon is infinite, there would be no salvage value term, and an alternative transversality condition is used:

$$\lim_{t \rightarrow \infty} e^{-r_i t} \lambda_i(t) = 0. \quad (7)$$

If the players do not use open-loop but feedback strategies, the necessary conditions for the costate variables become more complicated:

$$\dot{\lambda}_i = r_i \lambda_i - \frac{\partial H_i}{\partial x} - \sum_{j \neq i} \frac{\partial H_i}{\partial u_j} \frac{\partial \phi_j}{\partial x}, \quad \lambda_i(T) = \frac{\partial S_i}{\partial x}(x(T)). \quad (8)$$

The complication is that (7) involves the unknown strategies of the players and how those strategies depend on the state variables, and the system of equations (8), (1), is unsolvable.

It may be possible to determine feedback strategies by employing the Hamilton-Jacobi-Bellman equation argument of dynamic programming. Consider value functions  $V_i(t, x)$  for the players. A feedback equilibrium results if the following Hamilton-Jacobi-Bellman equations are satisfied:

$$r_i V_i - \frac{\partial V_i}{\partial t} = \max_{u_i \in U_i} \left[ g_i + \left( \frac{\partial V_i}{\partial x} \right)' f \right], \quad i = 1, \dots, n \quad (9)$$

with boundary conditions

$$V_i(T, x) = S_i(x), \quad i = 1, \dots, n. \quad (10)$$

For an infinite horizon,  $T = \infty$ , the  $\partial V_i / \partial t$  term on the left-hand side of (9) vanishes, and the boundary condition can be replaced by boundedness of the objective functionals (Dockner et al. [4]). Given the lack of a general theory of partial differential equations, it would appear that (9)–(10) would be difficult, if not impossible, to solve, which indeed is the case for many problems. However, for certain models it is possible to discern the functional form of the value functions, which allows a solution.

### 2.3. Time Consistency and Subgame Perfectness

There are two important requirements for Nash equilibria in dynamic noncooperative games: time consistency and subgame perfectness. Consider some  $(t, x) \in [0, T] \times X$ . A subgame  $\Gamma(t, x)$  is a differential game defined in the time interval  $(t, T)$  and having  $x(t) = x$  as an initial condition. In the subgame  $\Gamma(t, x)$ , player  $i$  has the objective functional

$$\int_t^T e^{-r_i(s-t)} g_i(s, x(s), u_1(s), \dots, u_n(s)) ds + e^{-r_i(T-t)} S_i(x(T)) \quad (11)$$

with system dynamics

$$\dot{x}(s) = f(s, x(s), u_1(s), \dots, u_n(s)), \quad x(t) = x. \quad (12)$$

Further,  $\Gamma(0, x_0)$  is the original game outlined in (1)–(3).

The requirement of *time consistency* is defined by the following. Let  $(\phi_1, \dots, \phi_n)$  be a Nash equilibrium of the game  $\Gamma(0, x_0)$  and let  $x(\cdot)$  be its unique equilibrium state trajectory. Suppose that for any  $t \in [0, T]$ , the subgame  $\Gamma(t, x)$  has a Nash equilibrium  $(\phi'_1, \dots, \phi'_n)$  such that  $\phi_i(s, y) = \phi'_i(s, y)$  for  $i \in \{1, \dots, n\}$  and for all  $(s, y) \in [t, T] \times X$ . Then, the Nash equilibrium  $(\phi_1, \dots, \phi_n)$  is time consistent. Open-loop Nash equilibria are time consistent, as are feedback Nash equilibria.

A stronger requirement is that of *subgame perfectness*, which is defined as follows. Let  $(\phi_1, \dots, \phi_n)$  be a Nash equilibrium of the game  $\Gamma(0, x_0)$ . Suppose that for any  $(t, x) \in$



$[0, T] \times X$ , the subgame  $\Gamma(t, x)$  has a Nash equilibrium  $(\phi'_1, \dots, \phi'_n)$  such that  $\phi_i(s, y) = \phi'_i(s, y)$  for  $i \in \{1, \dots, n\}$  and for all  $(s, y) \in [t, T] \times X$ . Then, the Nash equilibrium  $(\phi_1, \dots, \phi_n)$  is subgame perfect. If an equilibrium is subgame perfect, it is also time consistent.

The difference between time consistency and subgame perfectness is that subgame perfectness requires that an equilibrium also be an equilibrium for any subgame  $\Gamma(t, x)$ ,  $(t, x) \in [0, T] \times X$ , not just along the equilibrium state trajectory, the requirement for time consistency. A feedback Nash equilibrium is subgame perfect, but, unless it is feasible for the players to commit to unchangeable strategies at the outset of the game, an open-loop Nash equilibrium is not subgame perfect.

## 2.4. Example

An example, which is borrowed from Kamien and Schwartz [12], is offered to illustrate how open-loop and feedback Nash equilibrium strategies can be determined. Because the sections to follow describe applications involving advertising, the example here also serves as one involving other economic considerations. Consider that we have two competitors who produce an identical product, and need to develop strategies in terms of time-varying production schedules  $u_1(t)$  and  $u_2(t)$ . The cost of production is equivalent across the two competitors:

$$C(u_i) = cu_i + \frac{u_i^2}{2}, \quad i = 1, 2. \quad (13)$$

The competitors face a common price state variable  $p(t)$ , which changes according to the following dynamic relationship:

$$\dot{p} = s(a - u_1 - u_2 - p), \quad p(0) = p_0. \quad (14)$$

The parameter  $s$  measures the speed at which the price adjusts to the price determined by the demand function and total quantity supplied. Each firm chooses its output level  $u_i$  to maximize its discounted profit over an infinite horizon

$$\begin{aligned} J_i &= \int_0^\infty e^{-rt} (pu_i - C_i(u_i)) dt \\ &= \int_0^\infty e^{-rt} \left( pu_i - cu_i - \frac{u_i^2}{2} \right) dt, \quad i = 1, 2 \end{aligned} \quad (15)$$

subject to the dynamic constraint (14).

To find the open-loop Nash equilibrium for the game, first form the Hamiltonians

$$H_i = pu_i - cu_i - \frac{u_i^2}{2} + \lambda_i s(a - u_1 - u_2 - p), \quad i = 1, 2 \quad (16)$$

and obtain the necessary conditions

$$\frac{\partial H_i}{\partial u_i} = p - c - u_i - \lambda_i s = 0, \quad i = 1, 2 \quad (17)$$

and

$$\dot{\lambda}_i = r\lambda_i - \frac{\partial H_i}{\partial p} = (r + s)\lambda_i - u_i, \quad i = 1, 2 \quad (18)$$

also

$$\lim_{t \rightarrow \infty} e^{-rt} \lambda_i(t) = 0, \quad i = 1, 2. \quad (19)$$

Solving (17) for  $u_i$ , plugging into (18), integrating (18) with the help of an integrating factor, and using (19) to determine the constant of integration, results in

$$\lambda_i(t) = \int_t^\infty e^{-(2s+r)(\tau-t)} (p(\tau) - c) d\tau, \quad i = 1, 2. \quad (20)$$

It is clear from (20) that  $\lambda_1(t) = \lambda_2(t)$ . It follows from (17) that  $u_1(t) = u_2(t)$ . Recognizing this, subscripts  $i$  can be dropped from further derivation. Differentiating (17) with respect to time, substituting for  $\dot{p}$  from (14),  $\dot{\lambda}$  from (18), and  $\lambda$  from (17), yields

$$\dot{u} = (a + c)s + rc - (2s + r)p + ru. \quad (21)$$

With an infinite horizon, we can look for steady-state open-loop Nash equilibrium strategies, i.e., those for which  $\dot{u} = \dot{p} = 0$ . From (14) and (21), steady-state output and price are

$$\begin{aligned} u^{os} &= \frac{(a - c)(s + r)}{4s + 3r} \\ p^{os} &= \frac{2(a + c)s + (a + 2c)r}{4s + 3r}. \end{aligned} \quad (22)$$

To find feedback Nash equilibrium strategies, we use the value-function approach. The value function for each competitor is

$$rV_i = \max_{u_i} \left[ (p - c)u_i - \frac{u_i^2}{2} + \frac{\partial V_i}{\partial p} s(a - p - u_i - u_j) \right], \quad i = 1, 2, \quad i \neq j. \quad (23)$$

The maximization with respect to  $u_i$  yields

$$u_i = p - c - \frac{\partial V_i}{\partial p} s. \quad (24)$$

Substituting (24) into (23) produces

$$\begin{aligned} rV_i &= (p - c) \left( p - c - \frac{\partial V_i}{\partial p} s \right) - \frac{p - c - ((\partial V_i / \partial p) s)^2}{2} \\ &\quad + \frac{\partial V_i}{\partial p} s \left( a - 3p + 2c + s \left( \frac{\partial V_i}{\partial p} + \frac{\partial V_j}{\partial p} \right) \right), \quad i = 1, 2, \quad i \neq j. \end{aligned} \quad (25)$$

Note that in (25) the partial derivatives for both competitors appear. Interior solutions, i.e., those for which  $u_i > 0$ , are sought, and this is assured if we assume  $a > p_0$ . Solving the system of equations (25) means finding value functions  $V_1$  and  $V_2$  that satisfy them. The following quadratic functions are conjectured:

$$V_i(p) = A_i + B_i p + \frac{C_i p^2}{2}, \quad i = 1, 2 \quad (26)$$

which implies

$$\frac{\partial V_i}{\partial p} = B_i + C_i p, \quad i = 1, 2. \quad (27)$$

In order for the value functions (26) to be solutions to (25), the constants  $A_i$  and the coefficients  $B_i$  and  $C_i$  must have the certain values, which are found by substituting (26) and (27) into (25) to obtain

$$\begin{aligned} rA_i + rB_i p + \frac{rC_i p^2}{2} &= \frac{c^2}{2} + sB_i \left( \frac{sB_i}{2} + sB_j + a + 2c \right) \\ &\quad + (-c + s^2(B_i C_i + B_i C_j + B_j C_i) + sC_i(a + 2c) - 3sB_i)p \\ &\quad + \left( \frac{1}{2} - 3sC_i + s^2 C_i C_j + \frac{s^2 C_i^2}{2} \right) p^2, \quad i = 1, 2, \quad i \neq j. \end{aligned} \quad (28)$$

Now, equating the coefficients of  $p^2$  on both sides of the equation in (28) yields

$$s^2 C_i^2 + (2s^2 C_j - 6s - r)C_i + 1 = 0, \quad i = 1, 2, \quad i \neq j. \quad (29)$$

Equating the coefficients of  $p$  in (28) provides expressions for the  $B_i$  in terms of the  $C_i$ . Further, equating the constant terms in (28) gives expressions for the  $A_i$  as functions of the  $B_i$ . The next step is to establish that  $C_1 = C_2$ , from which it follows that  $B_1 = B_2$ ,  $A_1 = A_2$ , and that the feedback strategies of the two competitors are symmetric. To show that  $C_1 = C_2$ , subtract the two equations in (29) to obtain

$$(C_1 - C_2)(s^2(C_1 + C_2) - (6s + r)) = 0. \quad (30)$$

It follows that either  $C_1 = C_2$ , or

$$s^2(C_1 + C_2) = 6s + r. \quad (31)$$

Substituting from (27) into (24) and then into the state equation (14) yields

$$\dot{p} - sp(s(C_1 + C_2) - 3) = s(a + 2c + s(B_1 + B_2)). \quad (32)$$

The solution to (32) is

$$p(t) = \frac{a + 2c + s(B_1 + B_2)}{s(C_1 + C_2) - 3} + \left( p_0 - \frac{a + 2c + s(B_1 + B_2)}{s(C_1 + C_2) - 3} \right) e^{s(s(C_1 + C_2) - 3)t}. \quad (33)$$

In order for  $p(t)$  to converge as  $t \rightarrow \infty$ ,  $s(C_1 + C_2) < 3$  or  $s^2(C_1 + C_2) < 3s$  is required. However, from (31), this would require that  $3s + r < 0$ , which cannot hold because both  $s$  and  $r$  are nonnegative. This rules out the possibility of an asymmetric equilibrium. Having established that  $C_1 = C_2 = C$ , we can solve for roots from (29) to obtain

$$\bar{C}, \underline{C} = \frac{r + 6s \pm \sqrt{(r + 6s)^2 - 12s^2}}{6s^2}. \quad (34)$$

To select between the two roots, note again (33), from which it is seen that for  $p(t)$  to converge,  $2sC - 3 < 0$  or  $C < 3/2s$  is required. The larger root  $\bar{C}$  takes on its smallest value when  $r = 0$  in (34). But for  $r = 0$ ,  $\bar{C} = (3 + \sqrt{6})/3s > 3/2s$ . Thus, the larger root prevents convergence of  $p(t)$ . The smaller root  $\underline{C}$  achieves its highest value at  $r = 0$ , because  $\partial \underline{C} / \partial r < 0$ . At  $r = 0$ ,  $\underline{C} = (3 - \sqrt{6})/3s < 3/2s$ . Thus, only the smaller root allows for the convergence of  $p(t)$ . It also follows from (28) that

$$B = \frac{sC(a + 2c) - c}{r - 3s^2C + 3s}, \quad (35)$$

where  $C = \underline{C}$ , and

$$u^f(p) = (1 - sC)p - sB - c \quad (36)$$

is the feedback Nash equilibrium production strategy for each competitor. With the feedback strategies (36), the steady-state price from (14) is

$$p^{fs} = \frac{a + 2(c + sB)}{3 - 2sC} \quad (37)$$

and the steady-state output for each competitor is

$$u^{fs} = \frac{a(1 - sC) - (c - sB)}{3 - 2sC}. \quad (38)$$

It is evident, by comparing (37) and (38) to (22), that feedback strategies lead to different long-term outcomes than open-loop strategies do. The fact that feedback strategies differ from open-loop strategies is typically the case.

### 3. Differential-Game Applications

This section describes various applications of differential games to scenarios involving advertising competition. Providing the foundation for the applications are two important models of dynamic demand response to advertising: Lanchester (in Kimball [13]) and Vidale-Wolfe (in Vidale and Wolfe [15]), that were first introduced in consecutive issues of *Operations Research* in 1957.

Kimball [13] begins with Lanchester's formulation of the problem of combat, and develops four different models depicting the combat between two opposing forces. The fourth model, which since has become known as the Lanchester model, portrays the rate at which each force is able to capture its enemy's fighting units:

$$dn_1/dt = k_1n_2 - k_2n_1, \quad dn_2/dt = k_2n_1 - k_1n_2. \quad (39)$$

In (39),  $n_i$ ,  $i = 1, 2$ , is the number of units in force  $i$ 's possession at a moment in time, and  $k_i$  is the proportional rate at which  $i$  is able to capture its enemy's units  $n_j$ . The sum of units  $n_1 + n_2$  remains constant across time, but the individual values tend to steady-state values  $n_{1,0}$  and  $n_{2,0}$  such that  $n_{1,0}/n_{2,0} = k_1/k_2$ . Kimball reports that the model (39) has been used with great success to describe the effect of advertising, presumably in consulting contexts, and this has inspired researchers to investigate the model further.

Vidale and Wolfe [15] observe from large-scale advertising experiments that the interaction of sales and advertising can be described in terms of three parameters:

1. The Sales Decay Constant
2. The Saturation Level
3. The Response Constant.

Vidale and Wolfe [15] learn through the advertising tests that sales tend to decay in the absence of promotion, reach a saturation level in the presence of advertising, and respond to advertising in a way that can be characterized as a constant proportion per advertising dollar of the unsaturated portion of potential sales. They offer the following model to mathematically account for these phenomena:

$$dS/dt = rA(t)(M - S)/M - \lambda S, \quad (40)$$

where  $S$  is the rate of sales at time  $t$  and  $A(t)$  is the rate of advertising expenditure. The model (40) describes the dynamic sales response for a single product, but the model can be extended to a competitive situation, as the following application shows.

#### 3.1. Vidale-Wolfe Model Application

Deal [3] extends the Vidale-Wolfe model to a duopoly, as follows:

$$\begin{aligned} \dot{x}_1(t) &= -a_1x_1(t) + b_1u_1(t)[M - x_1(t) - x_2(t)]/M \\ \dot{x}_2(t) &= -a_2x_2(t) + b_2u_2(t)[M - x_1(t) - x_2(t)]/M, \end{aligned} \quad (41)$$

where  $x_i(t)$  = sales for brand  $i$  at time  $t$ ,  $u_i(t)$  = advertising expenditures for brand  $i$  at time  $t$ ,  $a_i$  = the sales decay parameter,  $b_i$  = the sales response parameter, and  $M$  = the total potential market size. Note that each competitor's advertising serves to attract sales not currently captured by either competitor. Also, sales lost through decay become part of the uncaptured potential.

To construct a differential game, Deal [3] considers the following performance index which combines profit over a finite planning horizon with market share at the end of the planning period:

$$\begin{aligned} \max_{u_1} J_1 &= w_1x_1(t_f)/[x_1(t_f) + x_2(t_f)] + \int_{t_0}^{t_f} \{c_1x_1(t) - u_1^2(t)\} dt \\ \max_{u_2} J_2 &= w_2x_2(t_f)/[x_1(t_f) + x_2(t_f)] + \int_{t_0}^{t_f} \{c_2x_2(t) - u_2^2(t)\} dt, \end{aligned} \quad (42)$$

where  $c_i$  = the net revenue coefficient for brand  $i$ ,  $w_i$  = the weighting factor for the performance index for brand  $i$ ,  $t_0$  = initial time of the planning horizon, and  $t_f$  = terminal time of the planning horizon. The maximization in (42) is subject to the dynamic constraints (41). To complete the differential game, Deal [3] specifies the constraints  $x_1(t_0) = x_{10}$ ,  $x_2(t_0) = x_{20}$ ,  $u_1(t) \geq 0$ ,  $u_2(t) \geq 0$ ,  $x_1(t) \geq 0$ ,  $x_2(t) \geq 0$ ,  $x_1(t) + x_2(t) \leq M$ .

Deal [3] derives open-loop Nash equilibrium solutions for the differential game. The Hamiltonians are:

$$\begin{aligned} H_1 &= c_1 x_1 - u_1^2 + \lambda_1(-a_1 x_1 + b_1 u_1(M - x_1 - x_2)/M) \\ &\quad + \lambda_2(-a_2 x_2 + b_2 u_2(M - x_1 - x_2)/M) \\ H_2 &= c_2 x_2 - u_2^2 + \gamma_1(-a_1 x_1 + b_1 u_1(M - x_1 - x_2)/M) \\ &\quad + \gamma_2(-a_2 x_2 + b_2 u_2(M - x_1 - x_2)/M). \end{aligned} \quad (43)$$

The necessary conditions for the solution are the system equations (41) with their initial conditions  $x_1(t_0) = x_{10}$  and  $x_2(t_0) = x_{20}$ , the costate equations and their corresponding terminal conditions

$$\begin{aligned} \dot{\lambda}_1 &= -c_1 + \lambda_1(a_1 + b_1 u_1/M) + b_2 u_2 \lambda_2/M, & \lambda_1(t_f) &= w_1 x_2(t_f)/(x_1(t_f) + x_2(t_f))^2 \\ \dot{\lambda}_2 &= \lambda_2(a_2 + b_2 u_2/M) + b_1 u_1 \lambda_1/M, & \lambda_2(t_f) &= -w_1 x_1(t_f)/(x_1(t_f) + x_2(t_f))^2 \\ \dot{\gamma}_1 &= \gamma_1(a_1 + b_1 u_1/M) + b_2 u_2 \gamma_2/M, & \gamma_1(t_f) &= -w_2 x_2(t_f)/(x_1(t_f) + x_2(t_f))^2 \\ \dot{\gamma}_2 &= -c_2 + \gamma_2(a_2 + b_2 u_2/M) + b_1 u_1 \gamma_1/M, & \gamma_2(t_f) &= w_2 x_1(t_f)/(x_1(t_f) + x_2(t_f))^2 \end{aligned} \quad (44)$$

and equations for the players' controls derived from  $\partial H_i / \partial u_i = 0$  for  $i = 1, 2$

$$\begin{aligned} u_1 &= b_1 \lambda_1(M - x_1 - x_2)/2M \\ u_2 &= b_2 \lambda_2(M - x_1 - x_2)/2M. \end{aligned} \quad (45)$$

The system equations (41) have initial conditions, and the costate equations (44) have terminal conditions. As such, the algorithm for finding the solution involves alternating passes forward and backward, with the state variables  $x_i$  being calculated on the forward passes, and the costate variables calculated on the backward passes. For the first forward pass, initial values for the control variables  $u_i$  for each discrete instant of time are guessed. In each backward pass, new values for the control variables  $u_i$  are calculated according to (45), and are used in the following forward pass. At the end of each forward pass, values for the performance indices are calculated, and when there has not been a significant change in the values, the algorithm is terminated.

Deal [3] uses the algorithm to test the following two hypotheses regarding advertising policies.

**Hypothesis 1.** *In a profit-maximizing situation, if the decay rate  $a_1 < a_2$  then, ceteris paribus, a dollar of advertising for Brand 1 will have a greater long-run effect than will Brand 2's advertising expenditures. The anticipated policy effect would be for Firm 1 to advertise more in order to maximize its performance index. This would be true particularly during the beginning of the planning period.*

**Hypothesis 2.** *If the weight placed on terminal market share  $w_1 < w_2$ , and all other competitive aspects of the model are identical, the advertising expenditure pattern for Brand 2 becomes one concerned with maximizing the terminal market share to the extent that actual losses might be sustained. The nature of the advertising curve under this objective changes from one that decreases to zero by  $t_f$  to one that increases sharply toward the terminal time.*

Numerical application of the algorithm confirms both hypotheses.

The Vidale-Wolfe model as formulated does not allow a feedback Nash equilibrium solution, although a modification of the model does allow a feedback solution, as is shown in Erickson [8].

### 3.2. Lanchester-Model Applications

The Lanchester model, the fourth model offered by Kimball [13], has been a popular model for studying advertising competition. The Lanchester model's appeal is that it captures the essence of one-on-one competitive rivalry. Erickson [5] uses the Lanchester model as a basis for studying pure market share rivalry in a market with fixed total sales, as well as a situation where sales expansion is possible.

The sales expansion model in Erickson [5] is

$$\dot{S}_i = f_{ij}S_j - f_{ji}S_i + f_i, \quad i = 1, 2, \quad i \neq j, \quad (46)$$

where  $S_i$  is competitor  $i$ 's sales level,  $f_{ij}$  is a general function of  $i$ 's advertising  $A_i$ , and  $f_i$  is a general function of  $A_i$ ,  $S_1$ ,  $S_2$ , and other exogenous factors such as growth in the economy and population. The competitors are assumed to choose advertising levels so as to maximize discounted cash flow over a finite time horizon, with no salvage value:

$$\max_{A_i} J_i = \int_0^T e^{-rt} (g_i S_i - A_i) dt, \quad i = 1, 2, \quad (47)$$

where  $r$  is a common discount rate and  $g_i$  is  $i$ 's unit contribution. Erickson pursues open-loop Nash equilibrium solutions. The Hamiltonians for the two competitors are:

$$\begin{aligned} H_1 &= g_1 S_1 - A_1 + L_{11}(f_{12}S_2 - f_{21}S_1 + f_1) + L_{12}(f_{21}S_1 - f_{12}S_2 + f_2) \\ H_2 &= g_2 S_2 - A_2 + L_{21}(f_{12}S_2 - f_{21}S_1 + f_1) + L_{22}(f_{21}S_1 - f_{12}S_2 + f_2), \end{aligned} \quad (48)$$

where the costate variables  $L_{11}$ ,  $L_{12}$ ,  $L_{21}$ ,  $L_{22}$  are subject to the following dynamic constraints under necessary conditions for an open-loop Nash equilibrium:

$$\begin{aligned} \dot{L}_{11} &= rL_{11} - g_1 + f_{21}(L_{11} - L_{12}) - L_{11}(\partial f_1 / \partial S_1) - L_{12}(\partial f_2 / \partial S_1), & L_{11}(T) &= 0 \\ \dot{L}_{12} &= rL_{12} - f_{12}(L_{11} - L_{12}) - L_{11}(\partial f_1 / \partial S_2) - L_{12}(\partial f_2 / \partial S_2), & L_{12}(T) &= 0 \\ \dot{L}_{21} &= rL_{21} - f_{21}(L_{22} - L_{21}) - L_{22}(\partial f_2 / \partial S_1) - L_{21}(\partial f_1 / \partial S_1), & L_{21}(T) &= 0 \\ \dot{L}_{22} &= rL_{22} - g_2 + f_{12}(L_{22} - L_{21}) - L_{22}(\partial f_2 / \partial S_2) - L_{21}(\partial f_1 / \partial S_2), & L_{22}(T) &= 0. \end{aligned} \quad (49)$$

For investigation of advertising and sales patterns with open-loop Nash equilibria, Erickson assumes the following functional forms:

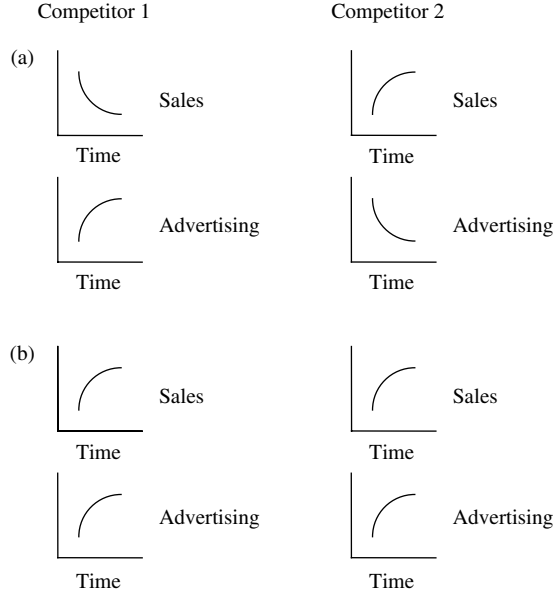
$$f_{12} = \beta_{12}A_1^{\alpha_1}, \quad f_{21} = \beta_{21}A_2^{\alpha_2}, \quad f_1 = \beta_1A_1^{\alpha_1}e, \quad f_2 = \beta_2A_2^{\alpha_2}e, \quad (50)$$

where  $e$  is a factor representing external growth potential, and uses random draws from the uniform distribution for each of the parameters and starting values of the sales levels, with the following ranges:

$r$ : 0.0–0.5  
 $g_1, g_2$ : 0–10  
 $\alpha_1, \alpha_2$ : 0.1–0.9  
 $\beta_1, \beta_2, \beta_{12}, \beta_{21}$ : 0.0–0.1  
 $e$ : 0.0–1.0  
 $S_1(0)$ : 0.0–1.0, with  $S_2(0)$  set to 1.0– $S_1(0)$ .

Fifty simulations are run, using an algorithm similar to Deal [3] to compute the advertising and sales levels. Two types of patterns emerged with frequency in the simulations, as shown in Figure 1. In Figure 1(a), advertising for one competitor shows growth whereas sales exhibit decline, and these tendencies reverse for the other competitor, essentially a pattern of sales adjustment between the competitors. In Figure 1(b), both advertising and sales for

FIGURE 1. Dynamic advertising and sales for a sales expansion model.



each competitor show growth, indicating that the competitors take advantage of significant growth potential in the market.

With a model of pure market share rivalry

$$\dot{M} = \beta_{12}A_1^{\alpha_1}(1 - M) - \beta_{21}A_2^{\alpha_2}M, \quad (51)$$

where  $M$  is competitor 1's market share, Erickson [5] develops analytical results. In steady state, relative advertising depends in the following way on the model parameters:

$$A_1/A = g_1\alpha_1/g_2\alpha_2. \quad (52)$$

Also, relative market share in steady state is as follows:

$$M/(1 - M) = (\beta_{12}/\beta_{21})(A_1/A_2)^{\alpha_2}A_1^{\alpha_1 - \alpha_2}. \quad (53)$$

If the elasticity parameters are equal,  $\alpha_1 = \alpha_2 = \alpha$ , we have

$$A_1/A_2 = g_1/g_2 \quad (54)$$

and

$$M/(1 - M) = (\beta_{12}/\beta_{21})(g_1/g_2)^{\alpha}. \quad (55)$$

Outside of steady state, that is, during the approach to steady state, Erickson shows that a competitor's advertising rises with time from a level low relative to the steady state ratio (52), whereas its rival's advertising declines from a relatively high level only if the competitor's market share is high relative to its steady-state ratio (53) and shrinking with time, a necessary condition. Conversely, market share being relatively low and increasing is a necessary condition for a competitor's advertising shrinking from a high level whereas its rival's advertising is increasing from a low level. This provides analytical support for the dynamic pattern seen in Figure 1(a), the pattern observed for simulations with a low external expansion value  $e$ , i.e., mature market situations.

Case [1] shows how what he terms *perfect* Nash equilibria can be derived for the Lanchester model, albeit a restricted version of the model. Case's perfect Nash equilibria are essentially stationary feedback Nash equilibria, but he generalizes the Hamilton-Jacobi-Bellman sufficient condition for such equilibria to

$$r_i V_i + C_i = \max_{u_i \in U_i} \left[ g_i + \left( \frac{\partial V_i}{\partial x} \right)' f \right], \quad i = 1, \dots, n, \quad (56)$$

where  $C_i$  is an arbitrary real constant.

Case's formulation of what he calls the two-player ad game is

$$\begin{aligned} 1 \quad & \text{maximize} \quad \int_0^\infty e^{-\rho t} [P \cdot x(t) - u^2(t)] dt \\ 2 \quad & \text{maximize} \quad \int_0^\infty e^{-\rho t} [Q - Q \cdot x(t) - v^2(t)] dt \\ & \text{subject to} \quad \dot{x}(t) = (1 - x(t))u(t) - \alpha x(t)v(t) \\ & \text{and} \quad 0 \leq x \leq 1, \quad u \geq 0, \quad v \geq 0, \end{aligned} \quad (57)$$

which is a differential game with Lanchester dynamics. For value functions  $V(x)$  for player 1 and  $W(x)$  for player 2, the generalized Hamilton-Jacobi-Bellman equation (56) yields:

$$\begin{aligned} u(x) &= (1/2)(1 - x)V'(x) \\ v(x) &= -(1/2)\alpha x W'(x) \end{aligned} \quad (58)$$

for advertising strategies, and

$$\begin{aligned} Px + (1/4)(1 - x)^2 V'^2(x) + (1/2)\alpha^2 x^2 V'(x)W'(x) &= \rho V(x) + c \\ Q(1 - x) + (1/2)(1 - x)^2 V'(x)W'(x) + (1/4)\alpha^2 x^2 W'^2(x) &= \rho W(x) + \gamma, \end{aligned} \quad (59)$$

which constitute a pair of ordinary differential equations. Now, solving (58) for  $V'(x)$  and  $W'(x)$ , integrating and substituting into (59) provides the following equations:

$$\begin{aligned} \frac{u^2}{2} - \left( \frac{\alpha x}{1 - x} \right) uv + \frac{Px}{2} - \rho \int^x \frac{u(s) ds}{1 - s} - \frac{c}{2} &= 0 \\ \frac{v^2}{2} - \left( \frac{1 - x}{\alpha x} \right) uv + \frac{Q(1 - x)}{2} + \frac{\rho}{\alpha} \int^x \frac{v(s) ds}{s} - \frac{\gamma}{2} &= 0. \end{aligned} \quad (60)$$

If  $\rho = 0$ , the equation system (60) can be solved. Case [1] states that solutions for small values of  $\rho$  are well approximated by those for  $\rho = 0$ . Define

$$R = \alpha x / (1 - x), \quad S = Px - c, \quad T = Q(1 - x) - \gamma. \quad (61)$$

The solution to (60), with  $\rho = 0$ , is

$$\begin{aligned} u(x) &= (1/\sqrt{3})(2TR^2 - S + 2(T^2R^4 - STR^2 + S^2)^{1/2})^{1/2} \\ v(x) &= (1/\sqrt{3})(2S/R^2 - T + 2(S^2/R^4 - ST/R^2 + T^2)^{1/2})^{1/2}. \end{aligned} \quad (62)$$

The equations (62) are well defined for  $0 < x < 1$ .

Case's [1] solution approach has limits. For one, it works only for a two-player game. For a three-player game, Case [1] illustrates that the Hamilton-Jacobi-Bellman equations become a system of simultaneous nonlinear partial differential equations, which is unsolvable analytically, a difficulty that exists for any game that involves more than one state variable.



Another limitation of the Case [1] approach is that restricting  $\rho$  to 0 is a severe constraint, especially for an infinite horizon problem.

Fruchter and Kalish [10] offer a different Nash equilibrium concept in order to deal with the limitations illustrated by Case [1]. Fruchter and Kalish [10] consider time-variant closed-loop strategies that depend on initial conditions. They define the objective functions for duopolistic competitors

$$\Pi_k(u_1, u_2) = \int_0^\infty [q_k x_k(t) - r_k u_k^2(t)] e^{-\mu t} dt, \quad k = 1, 2, \quad (63)$$

where

$$x_k = \begin{cases} x, & k = 1 \\ 1 - x, & k = 2 \end{cases} \quad (64)$$

and adopt the Lanchester model for market share dynamics

$$\dot{x}(t) = \rho_1[1 - x(t)]u_1(t) - \rho_2 x(t)u_2(t), \quad x(0) = x_0. \quad (65)$$

Using a variational approach, Fruchter and Kalish [10] obtain the first-order necessary conditions, where  $\lambda_k(t)$  is the Lagrange multiplier,

$$\dot{\lambda}_k(t) = \mu \lambda_k + (\rho_1 u_1(t) + \rho_2 u_2(t)) \lambda_k(t) + (-1)^k q_k, \quad \lim_{t \rightarrow \infty} \lambda_k(t) e^{-\mu t} = 0, \quad k = 1, 2 \quad (66)$$

and

$$u_k(t) = \frac{(-1)^{k+1}}{2} r_k^{-1} \rho_k \lambda_k(t) (1 - x_k(t)), \quad k = 1, 2. \quad (67)$$

Substituting (67) in (65) and (66) produces the following TPBVP:

$$\begin{aligned} \dot{x} &= \frac{1}{2} [r_1^{-1} \rho_1^2 (1 - x)^2 \lambda_1 + r_2^{-1} \rho_2^2 x^2 \lambda_2], \quad x(0) = x_0 \\ \dot{\lambda}_k &= \mu \lambda_k + \frac{1}{2} [r_1^{-1} \rho_1^2 \lambda_1 \lambda_k (1 - x) - r_2^{-1} \rho_2^2 \lambda_2 \lambda_k x] - (-1)^{k+1} q_k, \\ \lim_{t \rightarrow \infty} \lambda_k(t) e^{-\mu t} &= 0, \quad k = 1, 2. \end{aligned} \quad (68)$$

Using the notation  $x^P$  for  $x$  that satisfies (68), the TPBVP can be transformed into the following initial value problem (IVP):

$$\begin{aligned} \dot{x}^P &= \frac{1}{2} [r_1^{-1} \rho_1^2 (1 - x^P)^2 q_1 - r_2^{-1} \rho_2^2 (x^P)^2 q_2] e^{\mu t} \Phi(t), \quad x(0) = x_0 \\ \Phi'(t) &= \frac{1}{2} [r_1^{-1} \rho_1^2 q_1 (1 - x^P) + r_2^{-1} \rho_2^2 q_2 x^P] e^{\mu t} \Phi(t) - e^{-\mu t}, \quad \Phi(0) = \psi(x_0), \end{aligned} \quad (69)$$

where  $\psi(x_0)$  is obtained by solving the backward differential equation

$$\begin{aligned} \psi'(x^P) \psi(x^P) &[r_1^{-1} \rho_1^2 q_1 (1 - x^P)^2 - r_2^{-1} \rho_2^2 q_2 (x^P)^2] \\ &- [r_1^{-1} \rho_1^2 q_1 (1 - x^P) - r_2^{-1} \rho_2^2 q_2 x^P] \psi^2(x^P) = 2\mu \psi(x^P) - 2, \\ \lim_{t \rightarrow \infty} \psi(x^P(t)) e^{-\mu t} &= 0. \end{aligned} \quad (70)$$

Defining the closed-loop strategies

$$u_k^* = \frac{1}{2} r_k^{-1} \rho_k q_k \Phi(t) e^{\mu t} (1 - x_k), \quad k = 1, 2, \quad (71)$$

Fruchter and Kalish [10] prove that the pair  $(u_1^*, u_2^*)$  forms a global Nash equilibrium for the differential game. Because the equilibrium depends on initial conditions, it is not subgame perfect. They also remark that the closed-loop strategies use the same time-shape as the open-loop strategies

$$u_k^{OL} = \frac{1}{2} r_k^{-1} \rho_k q_k \Phi(t) e^{\mu t} (1 - x_k^P), \quad k = 1, 2. \quad (72)$$

Finally, Fruchter [9] extends the Fruchter and Kalish [10] closed-loop approach to an  $n$ -player game.

### 3.3. A Modified Lanchester Model

Subgame perfectness is a desirable feature for a Nash equilibrium to have, because it assures that the equilibrium strategies are in equilibrium for all possible values of the state variables, for any state the system may find itself in. The Fruchter and Kalish [10] closed-loop Nash equilibrium is not subgame perfect. Case [1] shows that a subgame perfect Nash equilibrium can be found for a Lanchester game involving a duopoly, but only when the market size is fixed, so that a single state variable, one competitor's market share, is involved, and for a discount rate of zero.

Sorger [14] offers a modification of the Lanchester model that permits the definition of subgame-perfect feedback Nash equilibria. The model variation is in the market share dynamics, the equation for which becomes

$$\dot{x}(t) = u_1(t)\sqrt{1-x(t)} - u_2(t)\sqrt{x(t)}, \quad x(0) = x_0. \quad (73)$$

The square-root terms  $\sqrt{1-x}$  and  $\sqrt{x}$  in (73) in effect brings diminishing returns into the state evolution process. Sorger [14] also argues that the state equation (73) approximates a model that includes a word-of-mouth effect.

Sorger [14] considers both a finite planning interval and an infinite horizon. For the finite interval, he considers the following objective functionals which the competitors wish to maximize:

$$\begin{aligned} J_1(u_1, u_2) &= \int_0^T e^{-r_1 t} [q_1 x(t) - (c_1/2)u_1^2(t)] dt + e^{-r_1 T} S_1 x(T) \\ J_2(u_1, u_2) &= \int_0^T e^{-r_2 t} [q_2 (1-x(t)) - (c_2/2)u_2^2(t)] dt + e^{-r_2 T} S_2 (1-x(T)). \end{aligned} \quad (74)$$

Sorger [14] derives both open-loop and feedback Nash equilibria for the differential game (73)–(74). The unique open-loop strategies are given by

$$\begin{aligned} u_1(t) &= (\eta_1(t)/c_1)\sqrt{1-y(t)} \\ u_2(t) &= (\eta_2(t)/c_2)\sqrt{y(t)}, \end{aligned} \quad (75)$$

where  $(\eta_1(t), \eta_2(t))$  is the unique solution to the differential equation system

$$\dot{\lambda}_i = r_i \lambda_i - q_i + \lambda_i^2/(2c_i) + \lambda_i \lambda_j/(2c_j), \quad \lambda_i(T) = S_i, \quad i = 1, 2, \quad i \neq j \quad (76)$$

and  $y(t)$  is the unique solution to the IVP

$$\dot{x} = (\eta_1(t)/c_1)(1-x) - (\eta_2(t)/c_2)x, \quad x(0) = x_0. \quad (77)$$

To obtain a feedback Nash equilibrium, Sorger [14] uses the Hamilton-Jacobi-Bellman approach, in which the value function for each competitor is linear in the state variable. The linearity of the value functions is due to the form of the modified Lanchester dynamic relationship (73). A feedback Nash equilibrium is given by the pair of strategies, where  $x_1 = x$  and  $x_2 = 1 - x$ ,

$$u_i(t, x) = (\eta_i(t)/c_i)\sqrt{1-x_i}, \quad i = 1, 2, \quad (78)$$

where the  $\eta_i(t)$  are the solution to

$$\dot{\lambda}_i = r_i \lambda_i - q_i + \lambda_i^2/(2c_i) + \lambda_i \lambda_j/c_j, \quad \lambda_i(T) = S_i, \quad i = 1, 2, \quad i \neq j. \quad (79)$$

Note that the system (79) differs from that in (76) for the open-loop solution. Sorger analytically compares the two equilibria, open-loop and feedback, and concludes that at least

one of the competitors applies a lower advertising effort in the feedback game compared to the open-loop game.

For an infinite time horizon, the salvage value is dropped. The unique open-loop Nash equilibrium is

$$u_1(t) = A_1 \sqrt{1 - y(t)}, \quad u_2(t) = A_2 \sqrt{y(t)}, \quad (80)$$

where  $(A_1, A_2)$  is the unique solution to

$$r_i A_i - q_i/c_i + A_i^2/2 + A_i A_j/2 = 0, \quad i = 1, 2, \quad i \neq j \quad (81)$$

satisfying  $A_i \geq 0$ ,  $i = 1, 2$ , and  $y(t)$  is given by

$$y(t) = A_1/(A_1 + A_2) + [x_0 - A_1/(A_1 + A_2)]e^{-(A_1 + A_2)t}. \quad (82)$$

A feedback Nash equilibrium is provided by the stationary feedback rules

$$u_i(t, x) = A_i \sqrt{1 - x_i}, \quad i = 1, 2, \quad (83)$$

where  $(A_1, A_2)$  is the unique solution to

$$r_i A_i - q_i/c_i + A_i^2/2 + A_i A_j = 0, \quad i = 1, 2, \quad i \neq j \quad (84)$$

satisfying  $A_i \geq 0$ ,  $i = 1, 2$ . For the infinite horizon situation, Sorger [14] shows that the advertising intensity is lower for both competitors in a feedback game than in an open-loop game. He also finds that a firm with a small initial market share would have a higher payoff in an open-loop equilibrium, whereas the firm with a high initial market share benefits from being in a feedback equilibrium.

## 4. Empirical Application

There has not been much empirical research conducted in the context of differential games. However, such empirical work is important to see if the differential game models and equilibrium implications can be useful in helping us understand the nature of advertising competition in real settings. This section offers an empirical application conducted by the author. In the application, a system of empirical equations is developed and estimated, a system that includes the representation of market share/sales dynamics as well as the endogenous determination of advertising expenditure levels according to feedback Nash equilibrium advertising strategies.

Erickson [6] uses the following Lanchester differential game as a framework for the empirical study of two duopolistic market settings:

$$i \max \int_0^\infty e^{-rt} (g_i M_i - A_i) dt, \quad i = 1, 2 \quad (85)$$

subject to  $\dot{M} = \beta_1 A_1^{\alpha_1} (1 - M) - \beta_2 A_2^{\alpha_2} M, \quad M(0) = M_0,$

where  $M_1 = M$  and  $M_2 = 1 - M$  in the objective functionals. The Case [1] equilibrium approach is used to derive advertising as functions of market share, which requires assuming that the discount rate  $r = 0$ . For the differential game (85), the equilibrium generates a system of relationships in which the advertising variables are defined implicitly in terms of market share:

$$g_i M_i + \frac{1 - \alpha_i}{\alpha_i} A_i - \frac{\beta_{3-i}}{\alpha_i \beta_i} A_i^{1-\alpha_i} A_{3-i}^{\alpha_{3-i}} \frac{M_i}{M_{3-i}} = c_i, \quad i = 1, 2. \quad (86)$$

The constants  $c_i$  in (86) are arbitrary real numbers. Explicit expressions of advertising as functions of market share can be obtained in the special case that the advertising elasticity parameters  $\alpha_i = 0.5$ , so that the market share equation is

$$\dot{M} = \beta_1 \sqrt{A_1} (1 - M) - \beta_2 \sqrt{A_2} M. \quad (87)$$

Defining

$$C_i \equiv \frac{M_i}{M_{3-i}}, \quad D_i \equiv \frac{\beta_i}{\beta_{3-i}}, \quad E_i \equiv g_i M_i - c_i, \quad i = 1, 2 \quad (88)$$

the advertising relationships become

$$A_i(M) = \frac{2(C_i^2/D_i^2)E_{3-i} - E_i + 2\sqrt{E_i^2 - (C_i^2/D_i^2)E_i E_{3-i} + (C_i^4/D_i^4)E_{3-i}^2}}{3}, \quad i = 1, 2. \quad (89)$$

Because the relationships representing equilibrium advertising strategies for the two competitors (86), (89) involve arbitrary constants  $c_i$ , the relationships are not unique; there are an infinite number of strategies that can qualify. Which particular strategies are actually adopted by the competitors becomes an empirical question, because the  $c_i$  can be considered parameters to be estimated through econometric analysis. A way of interpreting the constants is to note that in steady state the following holds:

$$c_i = g_i M_i - A_i, \quad i = 1, 2, \quad (90)$$

i.e., in steady state each firm's constant is equal to the net profit the firm is receiving. A way of interpreting the  $c_i$ , thus, is that they represent the profit that the competitors expect to be making once the market reaches steady state.

The two systems of relationships, (85)–(86) for the general model, and (87)–(89) for the constrained model, are systems of simultaneous nonlinear equations that are amenable to econometric analysis. Two empirical situations are studied. One involves the two leading soft drink brands Coca-Cola and Pepsi-Cola from 1968 through 1984, for which the constrained system (87)–(89) is assumed. Prior to estimating the full model, an empirical test is conducted to discern what kind of equilibrium strategies appear to be used by the soft drink competitors, open-loop or the Case [1] perfect equilibria. The Davidson and MacKinnon [2]  $P$  test of nonnested alternative nonlinear models is applied. For Coca-Cola, the hypothesis of an open-loop strategy is rejected in favor of a perfect equilibrium strategy, and the perfect equilibrium strategy is not rejected in favor of an open-loop strategy. For Pepsi-Cola, the results are more equivocal, in that neither hypothesis can be rejected in favor of the other. On balance, the evidence points to the Case [1] perfect equilibrium as providing a better explanation of the competing brands' advertising behavior.

Estimation of the system of simultaneous equations, assuming Case [1] perfect equilibrium advertising strategies on the part of the competing soft-drink brands, is accomplished with full-information maximum likelihood. Parameter estimates reveal the following insights:

- The brands are equally effective in their advertising.
- Coca-Cola is more profitable on a gross profit basis than is Pepsi-Cola.
- Coca-Cola is expected to have a higher profit level net of advertising than Pepsi-Cola in steady state.

Further, the steady-state market share implied by the model estimates is 0.59 for Coca-Cola, and the implied advertising strategies indicate that each competitor reacts to a deviation to its detriment from the steady-state market share with a sharp increase in its advertising.

The second empirical application of the Lanchester model is to the duopoly that existed in the U.S. beer market from 1971 through 1988 involving Anheuser-Busch and Miller. The general model of simultaneous relationships (85)–(86) is estimated with market share

TABLE 1. Estimates for Anheuser-Busch vs. Miller.

Parameter	Estimate	Standard error	$t$
$\alpha$	0.05102	0.01616	3.16
$\beta_1$	0.08633	0.04769	1.81
$\beta_2$	0.04221	0.02338	1.81
$g_1$	6,250	1,905	3.28
$g_2$	4,826	1,167	4.13
$c_1$	4,551	1,350	3.37
$c_2$	1,116	342	3.26

and advertising data on the two beer companies, although it is necessary to constrain the advertising elasticities to be equal to each other,  $\alpha_1 = \alpha_2 = \alpha$ . Full-information maximum likelihood provides the estimates in Table 1, which reveal that Anheuser-Busch is more effective with its advertising, has a higher gross-profit rate, and expects to have higher net profit in steady state, than Miller. In short, Anheuser-Busch appears to have an advantageous competitive position. Finally, advertising strategies for the two brewers implied by the model estimates show nonmonotonic advertising patterns as functions of market share.

## 5. Conclusions

This tutorial offers illustrations of the models and analysis methodologies that allow marketing science to study oligopolistic competition, in particular competition that involves advertising. It is not a comprehensive survey, in that certain differential game models are not covered, such as pricing models, advertising goodwill models, new product diffusion models, and models of marketing channels. Jørgensen and Zaccour [11], in particular, provide an excellent discussion of such models and other applications, and the reader is encouraged to consult that source.

We have made a good start in the study of dynamic marketing competition through the use of differential games. Much more needs to be done. The important challenge at the present juncture is to extend in an effective manner differential game modeling and analysis to general oligopolistic competition, and beyond duopolies. In this spirit, Erickson [8] offers a modified Vidale-Wolfe model of oligopolistic advertising competition, together with empirical applications of the model.

Competition and change are basic and powerful realities in market environments. With differential games, we are able to capture and learn from the essential dynamic and competitive aspects of markets.

## References

- [1] J. H. Case. *Economics and the Competitive Process*. New York University Press, New York, 1979.
- [2] R. Davidson and J. G. MacKinnon. Several tests for model specification in the presence of alternative hypotheses. *Econometrica* 49:781–793, 1981.
- [3] K. R. Deal. Optimizing advertising expenditures in a dynamic duopoly. *Operations Research* 27:682–692, 1979.
- [4] E. Dockner, S. Jørgensen, N. Van Long, and G. Sorger. *Differential Games in Economics and Management Science*. Cambridge University Press, Cambridge, UK, 2000.
- [5] G. M. Erickson. A model of advertising competition. *Journal of Marketing Research* 22:297–304, 1985.
- [6] G. M. Erickson. Empirical analysis of closed-loop duopoly advertising strategies. *Management Science* 38:1732–1749, 1992.
- [7] G. M. Erickson. *Dynamic Models of Advertising Competition*, 2nd ed. Kluwer Academic Publishers, Boston, MA, 2003.

- [8] G. M. Erickson. Advertising competition in dynamic oligopolies. Working paper, University of Washington Business School, Seattle, WA, 2007.
- [9] G. E. Fruchter. The many-player advertising game. *Management Science* 45:1609–1611, 1999.
- [10] G. E. Fruchter and S. Kalish. Closed-loop advertising strategies in a duopoly. *Management Science* 43:54–63, 1997.
- [11] S. Jørgensen and G. Zaccour. *Differential Games in Marketing*. Kluwer Academic Publishers, Boston, MA, 2004.
- [12] M. I. Kamien and N. L. Schwartz. *Dynamic Optimization: The Calculus of Variations and Optimal Control in Economics and Management*, 2nd ed. North-Holland, Amsterdam, The Netherlands, 1991.
- [13] G. E. Kimball. Some industrial applications of military operations research methods. *Operations Research* 5:201–204, 1957.
- [14] G. Sorger. Competitive dynamic advertising: A modification of the Case game. *Journal of Economic Dynamics and Control* 13:55–80, 1989.
- [15] M. L. Vidale and H. B. Wolfe. An operations-research study of sales response to advertising. *Operations Research* 5:370–381, 1957.

# Safe Scheduling

**Kenneth R. Baker**

Tuck School, Dartmouth College, Hanover, New Hampshire 03755,  
kenneth.r.baker@dartmouth.edu

**Dan Trietsch**

College of Engineering, American University of Armenia, Yerevan, Armenia,  
dan.trietsch@gmail.com

**Abstract** Safe scheduling augments the traditional approach to sequencing and scheduling models by explicitly recognizing safety times. In this tutorial, we give a brief review of the basic results in scheduling theory, describe how these results have been extended to stochastic cases, and demonstrate how safe-scheduling models sharpen our ability to formulate stochastic problems. Although scheduling theory has historically been driven by optimization methods, we demonstrate how stochastic scheduling problems can be approached with a powerful spreadsheet-based heuristic procedure.

**Keywords** sequencing; stochastic scheduling; safety time

---

Safe scheduling takes a new approach to stochastic problems in scheduling by explicitly recognizing safety times. In this tutorial, we discuss the basic principles and results of safe scheduling. We begin with a brief historical review of deterministic scheduling theory. From that starting point, we describe how the theory expanded to stochastic models. Finally, we show how safe-scheduling models fit into the picture and improve our ability to address stochastic scheduling problems.

## 1. Background: The Basic Deterministic Model

In scheduling theory, generic models are built to represent practical scheduling problems, or at least abstractions of practical problems, and those models are analyzed in an attempt to discover general rules and principles of effective scheduling. The most basic and most important model in scheduling theory is the deterministic, single-machine model. It is based on the following set of assumptions.

- A1. At time zero,  $n$  single-operation jobs (or tasks) are available for processing.
- A2. The machine (or resource) can process at most one job at a time.
- A3. Setup times for the jobs are independent of job sequence and are included in processing times.
- A4. Job descriptors are deterministic and known in advance.
- A5. Machines are continuously available (no breakdowns occur).
- A6. Machines are never kept idle while work is waiting.
- A7. Once an operation begins, it proceeds without interruption.

For the basic model, we may not need to impose the last two assumptions. In most deterministic, single-machine models, it is possible to find optimal solutions without inserted idle time, as in A6, or job preemption, as in A7. Therefore, A6 and A7 emerge as properties of optimal solutions rather than as assumptions that make the model tractable. In more complex models, however, these two properties may not characterize optimal solutions, so it is helpful to keep them visible.

To formalize the elements of the basic model, let  $p_j$  denote the time required to process job  $j$ . (In some cases, we might also want to specify the job's due date,  $d_j$ , or the job's weighting factor,  $w_j$ .) When we construct a schedule, consisting of a sequence of the jobs, we effectively determine the completion time of each job,  $C_j$ . Next, we want to convert the set of completion times into a performance measure for the schedule.

A performance measure is a one-dimensional summary that aggregates the results for the various jobs. For example, suppose we define the *flowtime*  $F_j$  as the length of time that job  $j$  is in the system until it is fully processed. In the basic model, of course, we have  $F_j = C_j$ , but in other models the two measures could differ. A common way to aggregate over all the jobs is to take the total or the average; thus, we are interested in minimizing the total of the flowtimes, or  $F = \sum F_j$ . We refer to this as the  $F$ -problem. When we want to distinguish importance weights,  $w_j$ , among the various jobs, we can generalize the total flowtime measure to the *total weighted flowtime* measure, or  $\sum w_j F_j$ . This variation of the performance measure gives rise to what we call the  $F_w$ -problem.

When due dates,  $d_j$ , are given and we are concerned about due-date performance, we begin by measuring the tardiness of job  $j$ ,  $T_j = \max\{0, C_j - d_j\}$ . Tardiness measures the amount by which a job's due date is missed, so one alternative for quantifying due-date performance is the total tardiness measure or  $\sum T_j$ . Alternatively, if there is a good chance that all jobs can be completed on time, or if we are mainly concerned about the worst-case outcome, we can adopt the *maximum tardiness* measure, or  $T_{\max} = \max\{T_j\}$ , and try to sequence the jobs so that its value is zero.

Finally, when we are mainly concerned about the efficiency of the schedule, we might want to complete all jobs as soon as possible, and so adopt the *makespan* measure, or  $C_{\max} = \max\{C_j\}$ . However, in the basic model, this performance measure does not help us to distinguish among schedules, because all sequences have the same makespan. The makespan measure becomes challenging for more complicated models.

Although scheduling problems can be classified in different ways, one of the most useful classifications is based on the scheduling objective. Three main objectives are prominent in scheduling theory, just as they are in scheduling practice:

- Turnaround
- Due-date performance
- Throughput

In scheduling models, these objectives are represented by such quantitative performance measures as  $\sum w_j F_j$ ,  $T_{\max}$ , and  $C_{\max}$ , and the models with these objectives are analyzed to discover insights and relationships that might inform practical scheduling decisions. When scheduling theorists discuss the origins of their field, they usually point to three seminal papers that introduced these three types of criteria. Smith [11] addressed the single-machine sequencing problem with the objective of minimizing  $\sum w_j F_j$ . He demonstrated the following property.

**Theorem 1.** *To minimize  $\sum w_j F_j$  in the basic single-machine model, sequence the jobs in nondecreasing order of  $p_j/w_j$ . This is often called the shortest weighted processing time (SWPT) sequence. If all weights are identical, this result dictates processing the jobs in shortest-first order, a procedure that generates the so-called shortest processing time (SPT) sequence.*

Jackson [5] dealt with the same model and the  $T_{\max}$  objective. He demonstrated the following result.

**Theorem 2.** *To minimize  $T_{\max}$  in the basic single-machine model, sequence the jobs in nondecreasing order of  $d_j$ . This is often called the earliest due date (EDD) sequence.*

Johnson [6] analyzed a two-machine model with the objective of maximizing throughput. Specifically, for a given set of tasks, throughput is maximized when all jobs are completed in the shortest possible time. Johnson derived the following rule for sequencing the jobs.



**Theorem 3.** *To minimize  $C_{\max}$  in the two-machine flowshop (where all jobs must travel first to machine A and then to machine B), first classify the jobs according to whether the processing time on machine A is smaller than the time on machine B. Then, sequence the jobs with smaller times on A in nondecreasing order of processing times on machine A, followed by the jobs with smaller times on B in nonincreasing order of processing time on machine B.*

Theorems 1–3 represent key results in the seminal work done more than 50 years ago. Subsequent research in scheduling theory led to additional results of interest, which we highlight next.

Let  $U_j$  denote a binary measure that is 1 if job  $j$  is tardy and 0 otherwise. Consider the problem of minimizing the number of tardy jobs,  $U = \sum U_j$ , which we refer to as the  $U$ -problem. Moore [8] described a relatively simple solution algorithm that constructs two sets of jobs—an early set  $A$ , which may be sequenced according to EDD, and a late set  $R$ , which may be sequenced in any order. Moore’s algorithm proceeds as follows.

1. Place all jobs in  $A$  and order the jobs according to EDD.
2. If no jobs are tardy, then stop. The sequence is optimal.
3. Consider the jobs in sequence up to and including the first tardy job. Among the jobs in this initial sequence, remove the longest from  $A$  and place it in  $R$ . Return to Step 2.

At the end of this procedure, the jobs in  $A$  will remain in EDD order and complete by their due dates. The jobs in  $R$  may be sequenced in any order because they will be tardy.

Several research efforts were devoted to the problem of minimizing total tardiness,  $T = \sum T_j$ , which we refer to as the  $T$ -problem. Although this sequencing problem exhibits special structure, and many special cases can be solved readily, the problem is NP-hard in general. Over the years, many specialized algorithms have been developed for the  $T$ -problem, and the state of the art now reflects methods that can solve problems with as many as a few hundred jobs (Szwarc et al. [13]).

The advent of just-in-time production generated interest in performance measures based on both earliness and tardiness. A job’s earliness is defined as  $E_j = \max\{0, d_j - C_j\}$ , which becomes a symmetric complement of the job’s tardiness. The E/T-problem calls for minimizing the sum of earliness costs and tardiness costs. For example, let  $\alpha_j$  denote the unit penalty for earliness and let  $\beta_j$  denote the unit penalty for tardiness. Then the E/T objective function takes the form of the sum  $\sum(\alpha_j E_j + \beta_j T_j)$ . There are many special cases of the E/T-problem, depending on whether the due dates are different and whether the earliness and tardiness penalties differ, but the general E/T-problem is NP-hard.

The E/T-problem differs in an important way from the other problems we have mentioned. The performance measures  $\sum w_j F_j$ ,  $T_{\max}$ ,  $\sum U_j$ , and  $\sum T_j$  are all *regular* measures. For a regular measure to improve, at least one job must complete earlier. Stated another way, if we modify a schedule by delaying one job’s completion time without reducing some other completion time, then the measure cannot get better. Most familiar performance measures are regular measures. Single-machine models with regular measures have optimal solutions without inserted idle time or preemption. For that class of problems, therefore, assumptions A6 and A7 are redundant. The E/T-problem, by contrast, involves a nonregular measure. As a result, we may sometimes impose assumptions A6 and A7, but in doing so, we may be excluding some desirable scheduling possibilities. Nevertheless, assumptions A6 and A7 are very practical—schedules and schedulers adhere to these assumptions in the real world—so those assumptions are usually reasonable in the analysis of scheduling problems.

### 1.1. A Spreadsheet-Based Heuristic Solution Approach

Difficult problems such as the E/T-problem or the  $T$ -problem generally require specialized algorithms and codes for finding optimal solutions to anything more than modest-sized problems. Nevertheless, good heuristic procedures are available for solving these problems. In

TABLE 1. Spreadsheet layout for a 20-job tardiness problem.

Sequencing model																				
Single-machine tardiness example																				
Data																				
$j$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$p_j$	60	71	71	76	81	82	93	104	108	108	108	109	113	115	116	118	118	120	122	145
$d_j$	404	394	534	308	778	917	482	472	702	803	1,142	1,115	811	1,191	672	1,139	1,329	710	534	591
Solution																				
5,433																				
Decisions and calculations																				
Pos	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$j$	2	7	4	1	8	3	19	9	5	10	6	13	11	12	14	15	16	17	18	20
$p_j$	71	93	76	60	104	71	122	108	81	108	82	113	108	109	115	116	118	118	120	145
$C_j$	71	164	240	300	404	475	597	705	786	894	976	1,089	1,197	1,306	1,421	1,537	1,655	1,773	1,893	2,038
$d_j$	394	482	308	404	472	534	534	702	778	803	917	811	1,142	1,115	1,191	672	1,139	1,329	710	591
$T_j$	0	0	0	0	0	0	63	3	8	91	59	278	55	191	230	865	516	444	1,183	1,447

what follows, we show how a generic heuristic procedure can be implemented on a spreadsheet for the purposes of solving these kinds of difficult problems. We use the  $T$ -problem as an example.

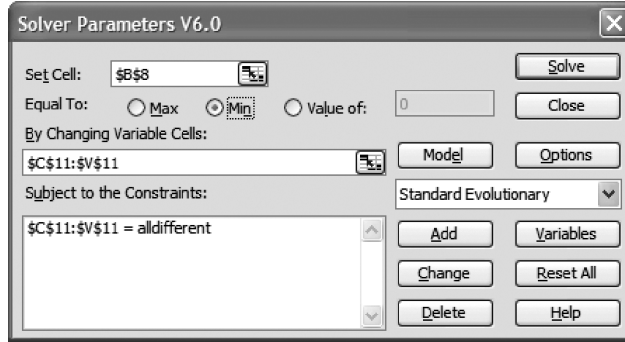
Imagine that we were given a single-machine sequencing problem and asked to verify the objective function value corresponding to a particular solution. How would we organize the calculations? The given information can be displayed as a table of job numbers, processing times and due dates, one type of data in each of three rows. The sequence can be displayed in another row. For each position in sequence, we would note the job assigned to that position and its processing time, compute its completion time, compare that with its due date, and finally, calculate its tardiness. The complete layout for an example is illustrated in the spreadsheet of Table 1.

In the spreadsheet, rows 4–6 contain the given data that describe the problem. Rows 10–15 contain the calculations required to evaluate the objective function, and the value of the objective function itself is shown in row 8. Note, for example, that the table of calculations contains some entries that are drawn from the table of given information. In a spreadsheet context, these entries are table lookups and are obtained via Microsoft Excel formulas. The job completion times are easily calculated from left to right with a cumulative sum. Finally, the tardiness values are calculated from the formula that defines tardiness,  $\max\{0, \textit{Completion} - \textit{due date}\}$ , and these are totalled in the performance measure cell.

The optimization problem corresponding to our example is to choose the sequence in row 11 to minimize the value in the performance measure cell in row 8. An effective way of solving such a problem in Excel is to implement the evolutionary algorithm in Premium Solver. (Premium Solver is an upgraded version of the solver that comes with Excel and is available from Frontline Systems, <http://www.solver.com>. An educational version is common in engineering and business curricula because it accompanies more than a dozen textbooks in widespread use. Premium Solver contains four different algorithms, one of which is Evolutionary Solver, which is an advanced genetic algorithm suitable, among many other things, for sequencing problems.)

When we invoke Premium Solver, we encounter the Solver window shown in Figure 1. We select Evolutionary Solver from the pulldown menu, and the remaining entries specify (1) a cell containing the objective function, (2) a range of cells containing the decisions—in this case the assigned positions in sequence, and (3) any constraints on the selection of the decision variables. Here, the one constraint takes the form of an *all-different* constraint. This requirement forces the positions in the decision variable cells to comprise a permutation of the job indices. In other words, the decision cells must correspond to a feasible sequence.

FIGURE 1. Solver parameters window.



Evolutionary Solver searches for the best solution it can find, and its effectiveness is influenced by several user-determined parameters that are specified after clicking the Options button in the Solver window. The most important of these parameters set the stopping and convergence conditions that control the termination of the search. A good generic set of parameters would be the following:

- Population size = 100
- Mutation rate = 7.5% (the default value)
- Convergence = 0.01%
- Tolerance = 0
- Maximum time without improvement = 15 seconds
- Maximum search time = 60 seconds.

Other parameters (the number of feasible solutions, the number of subproblems, and the number of iterations) should be set to large numbers so as not to impede the search.

With these settings, the search will continue until one of the stopping conditions is met:

- The search has been underway for 60 seconds;
- No improvement has been encountered in the last 15 seconds;
- 99% of the 100 best solutions found are within 0.01% of each other.

The time limits can be adjusted according to the user's patience, but we have found that runs of roughly one minute produce good results for sequencing problems. In fact, optimal or near-optimal solutions are usually found in far less time.

In our 20-job example, a run of Evolutionary Solver produces the schedule shown in Table 1, with a total tardiness of 5,433. (This turns out to be an optimal value.)

## 2. Traditional Models in Stochastic Scheduling

Practical scheduling insights can be gleaned from relaxing any of the assumptions in the basic model and analyzing the problem that arises. Thus, for example, if we relax A1 and allow jobs to arrive over time, we confront the dynamic version of the basic single-machine model. If we relax A3 and allow changeover times that depend on job sequence, we confront the sequence-dependent setup version of the basic model. Each relaxation of this type produces a class of interesting scheduling problems, but space does not permit us to summarize the work on all of those models. For our purposes, we focus on the relaxation of A4, the assumption of deterministic job parameters.

When processing times are random, the resulting problem is called a *stochastic scheduling problem*. By relaxing A4, we permit due dates and job weights to be uncertain as well as processing times, but such models have limited practical significance. The most practical version of the model contains probabilistic processing times but treats due dates and weights as deterministic. As a consequence, the EDD sequence is well defined (except for ties). In

contrast, the SPT sequence is not well defined, because processing times are not known in advance. However, it is still possible to order jobs by nondecreasing *expected* processing times. This sequence is known as Shortest Expected Processing Time (SEPT). Similarly, the Shortest Weighted Expected Processing Time sequence (SWEPT) is also well defined.

Historically, research on stochastic scheduling has focused on the same performance measures considered in deterministic scheduling ( $F$ ,  $T_{\max}$ ,  $U$ ,  $T$ , etc.) and has sought the minimization of their expected values. Thus, typical stochastic models aim to minimize  $E(F)$ ,  $E(T_{\max})$ ,  $E(U)$ ,  $E(T)$ , etc. We refer to such models as *stochastic counterparts* of the corresponding deterministic problems. For example, the stochastic counterpart of the  $F$ -problem is a stochastic scheduling problem in which the objective function is the expected total flow-time,  $E(F)$ . More generally, for deterministic models that seek to minimize the total cost or the maximum cost, stochastic counterparts seek to minimize the expected total cost or the expected maximum cost.

Advances in scheduling theory have provided solutions to some of the prominent stochastic counterparts, but such results are often difficult to obtain. Before exploring those results, we discuss a general way to formulate stochastic sequencing problems (using spreadsheet models) along with a general heuristic procedure for obtaining solutions.

In formulating a stochastic model, the key feature is a description of the distributions of processing times. Imagine a table in which each row corresponds to a scenario and each column corresponds to a job. The body of the table contains the processing time of job  $j$  in scenario  $s$ . In Table 2, we show an example with four scenarios and five jobs. Along the bottom of the table, we show the average processing time for each job.

We can interpret this table in two ways. First, we can think of the scenarios as states of nature. When the scenarios form an exhaustive set of states, the columns represent the population of processing time outcomes for each job. Based on those values, we could calculate the mean and variance of processing times as well as the mean and variance of virtually any measure of performance. These calculations would provide us with accurate theoretical values.

Second, we can think of the scenarios as observations drawn from the population of potential outcomes. In this situation, the columns represent samples from processing-time distributions. Based on those values, we could calculate sample averages and sample standard deviations, and we could construct estimates of any measure of performance. Moreover, statistical procedures would allow us to estimate the precision in those estimates. In this *sample-based* approach, we usually find that samples of size 1,000 are sufficient for our purposes.

In Table 3, we show a spreadsheet formulation of a small stochastic scheduling problem. This example uses the probability distributions of Table 2. Because there are four scenarios, there are four rows in each portion of the table corresponding to the various probabilistic outcomes for completion time, tardiness, etc. For the purposes of illustration, we treat the scenarios in this example as if they represent an exhaustive set of probabilistic outcomes, so that the average values correspond to the true mean of the distributions involved. Suppose

TABLE 2. Scenarios for a stochastic scheduling problem.

Example problem with random processing times					
Scenario	Job 1	Job 2	Job 3	Job 4	Job 5
1	10	8	12	15	20
2	12	12	14	17	20
3	14	16	16	20	20
4	16	20	22	24	24
Average	13	14	16	19	21

TABLE 3. Spreadsheet layout for the example problem.

Example	$U$ -problem						
Data	Job $j$	1	2	3	4	5	
State	$d_j$	22	37	28	75	54	
1	$p_j$	10	8	12	15	20	
2		12	12	14	17	20	
3		14	16	16	20	20	
4		16	20	22	24	24	
	$E(p_j)$	13	14	16	19	21	
		No. tardy					
	Sequence	1	3	2	5	4	2.50
Processing times	1	10	12	8	20	15	
	2	12	14	12	20	17	
	3	14	16	16	20	20	
	4	16	22	20	24	24	
Completion times	1	10	22	30	50	65	
	2	12	26	38	58	75	
	3	14	30	46	66	86	
	4	16	38	58	82	106	
Tardiness	1	0	0	0	0	0	
	2	0	0	1	4	0	
	3	0	2	9	12	11	
	4	0	10	21	28	31	
		Total					
Unit tardiness	1	0	0	0	0	0	0
	2	0	0	1	1	0	2
	3	0	1	1	1	1	4
	4	0	1	1	1	1	4

that the objective function is  $E(U)$ , the expected number of tardy jobs. As shown in Table 3, the EDD sequence 1-3-2-5-4 generates a value of 2.50.

Again, we can employ Evolutionary Solver to find good solutions for this type of problem. The objective function cell (H12 in the table), and the decision cells (C12:G12) represent the heart of the optimization problem. The only constraint is the all different constraint on the decisions. A run of Evolutionary Solver generates a solution value of 1.50, produced by the sequence 1-2-5-4-3.

Evolutionary Solver gives us a powerful and flexible solution technique that can run effectively with up to 20 jobs and up to 1,000 scenarios. Although the technique is a heuristic method and cannot guarantee optimal solutions, it seems to be very effective in problems of this size.

### 2.1. Optimal Solutions for Stochastic Scheduling

Of course, we would prefer to work with solutions that are guaranteed to be optimal in stochastic scheduling problems, but it turns out that there are very few general cases in which we can do so. We now turn to a brief summary of the theoretical state of the art in stochastic scheduling. We focus on the measures addressed earlier:  $E(F_w)$ ,  $E(T_{\max})$ ,  $E(U)$ , and  $E(T)$ , and we make no specific assumptions about the processing-time distributions.

Consider the stochastic counterpart of the  $F_w$ -problem. In other words, processing times are random, and the objective is to minimize the expected value of total weighted flowtime.

**Theorem 4** (Rothkopf [10]). *To minimize  $E(F_w)$  in the basic single-machine model, sequence the jobs according to SWEPT.*

Note that Theorem 4 does not say that total flowtime is *always* minimized by SWEPT—that is, in every scenario. Rather, SWEPT minimizes the total *on average*. In the scenarios of Table 2, sequences other than SWEPT are sometimes optimal. In the first scenario, for example, job 2 should precede job 1, and in the fourth scenario, job 3 should precede job 2. But such observations can be made only in hindsight, and we cannot rely on hindsight for scheduling decisions. Therefore, SWEPT is the best we can do *ex ante*, before the realizations of the processing times are revealed.

In the example of Table 2, all weights are equal to 1, so the objective function is total expected flowtime. By Theorem 4, the optimal sequence is SEPT (1–2–3–4–5), which generates an expected total flowtime of 228. Suppose instead that we substitute the expected processing times for the random processing times and then solve this deterministic problem, thus creating what is known as the *deterministic counterpart* of the stochastic problem. We then find that the same sequence (1–2–3–4–5) is optimal, and that its total flowtime is 228. In general, the optimal sequence and the optimal value of the performance measure in the deterministic counterpart always match those of the stochastic  $F_w$ -problem.

Next, consider the stochastic counterpart of the  $T_{\max}$ -problem. The solution to this problem is well known, and again, it echoes the deterministic solution.

**Theorem 5** (Crabill and Maxwell [4]). *To minimize  $E(T_{\max})$  in the basic single-machine model, sequence the jobs according to EDD.*

But an interesting feature arises with respect to the deterministic counterpart. When we suppress the random processing times and substitute expected values instead, the optimal sequence is the same (EDD does not depend on the processing times). However, the value of the objective function may not be the same for the deterministic counterpart as it is for the solution of the stochastic problem. Our example serves as an illustration. Table 4 shows that the EDD sequence produces  $E(T_{\max}) = 11.75$ . But if we substitute expected processing

TABLE 4. The EDD sequence in the example problem.

Example	$T_{\max}$ -problem						
Data	Job $j$	1	2	3	4	5	
State	$d_j$	22	37	28	75	54	
1	$p_j$	10	8	12	15	20	
2		12	12	14	17	20	
3		14	16	16	20	20	
4		16	20	22	24	24	
	$E(p_j)$	13	14	16	19	21	
							$E(T_{\max})$
	Sequence	1	3	2	5	4	11.75
Processing times	1	10	12	8	20	15	
	2	12	14	12	20	17	
	3	14	16	16	20	20	
	4	16	22	20	24	24	
Completion times	1	10	22	30	50	65	
	2	12	26	38	58	75	
	3	14	30	46	66	86	
	4	16	38	58	82	106	
							$T_{\max}$
Tardiness	1	0	0	0	0	0	0
	2	0	0	1	4	0	4
	3	0	2	9	12	11	12
	4	0	10	21	28	31	31

times for random processing times and solve the deterministic problem that results, we find that the EDD sequence produces  $T_{\max} = 10.00$ . Thus, we begin to see that we cannot go back and forth seamlessly between a stochastic-scheduling problem and a deterministic simplification based on expected values.

The mechanism operating in this example is an instance of Jensen's Inequality:

$$\text{If } h(Z) \text{ is convex and } Z \text{ is random, then } E[h(Z)] \geq h[E(Z)].$$

For example, taking  $Z$  to be a set of processing times and  $h(Z)$  to be the maximum tardiness function, we obtain  $E[\max(T_j)] \geq T_{\max}$  in the deterministic counterpart. Therefore, we can conclude that, although it delivers the optimal sequence to the stochastic problem, the deterministic counterpart systematically underestimates the value of the objective function in the stochastic problem.

In such a case, we say that there is a nonnegative *Jensen gap* between the objective function of the deterministic counterpart and the objective of the stochastic problem. Jensen gaps are often positive, although they are zero for linear functions, such as  $F_w$ . To distinguish between piecewise linear convex functions (such as the max function) and linear ones, we define any convex function that is not linear as *meaningfully convex*. In models with meaningfully convex objective functions, we can find instances with strictly positive Jensen gaps. If we are interested in the optimal value of the objective function as well as the optimal sequence, the deterministic counterpart may be biased, due to the Jensen gap, even in otherwise well-solved problems.

When we consider the stochastic counterpart of the  $U$ -problem, things get a little more complicated. Although the deterministic counterpart can be solved efficiently using Moore's algorithm, the stochastic  $U$ -problem is essentially unsolved. In other words, the deterministic counterpart may not even reveal the optimal sequence to the stochastic problem. We return to this model later on.

Next, consider the stochastic  $T$ -problem. The deterministic counterpart has attracted a lot of attention over the years, and as mentioned earlier, the most recent algorithmic advances can solve versions of the problem with hundreds of jobs. However, there has been little progress on the stochastic problem. One reason is the difficulty of finding dominance conditions, which are often the key to effective performance of deterministic solution procedures. As a consequence, the branch-and-bound approaches that work well in the deterministic case have not been successful in the stochastic case. (Because the tardiness function is meaningfully convex, we should not be surprised to discover that a Jensen gap exists here, too.) The stochastic  $T$ -problem is therefore largely unsolved.

To reinforce our point, consider the problem of sequencing two jobs with stochastic processing times and with the objective of minimizing the expected total tardiness  $E(T)$ .

Job $j$		1	2
$d_j$		2	3
$E(p_j)$		2.5	2.5

State	Job $j$	1	2	Probability
A	$p_j$	1	2	0.5
B	$p_j$	4	3	0.5

We can easily enumerate the possible solutions to this problem. The sequence 1–2 generates  $E(T) = 4.5$ , whereas the sequence 2–1 generates  $E(T) = 3.5$ . Therefore, the latter sequence is optimal. However, instead of analyzing the stochastic problem, suppose we rely on the deterministic counterpart, in which the expected processing times are 2.5 for each job. When

we substitute this value for the processing times, we find that the minimum tardiness is 2.5, achieved by the sequence 1–2. Thus, the deterministic counterpart leads us to a suboptimal sequence as well as to a biased value for the objective function.

Thus, as we review the general results in stochastic scheduling, we find that the deterministic counterpart is not reliable. The bias inherent in such an approach has a potentially devastating consequence for practice. Over time, as organizations learn that their deterministic counterpart solutions underestimate actual performance, they sense the need for time buffers in the schedule. Lacking any explicit guidance for buffering schedules, practitioners tend to make pessimistic estimates of processing times. This approach is ultimately inefficient. For practical solutions of stochastic scheduling problems, we must turn to a different form of analysis. Heuristic solutions are available, using the sample-based technique described above, but optimization approaches require a new way of thinking about the stochastic model.

### 3. The Safe-Scheduling Approach

We believe the extensive literature on stochastic scheduling models has unfortunately focused too often on stochastic counterparts and overlooked the practical importance of safety time. As a result, traditional stochastic models fail to consider service levels in scheduling. We advocate an approach to the stochastic problem that recognizes safety time. As an analogy, imagine inventory analysts trying to build stochastic inventory models by relying only on deterministic analysis of average behavior and making no provisions for safety stock. Just as safety *stocks* are vital to practical inventory policies, safety *time* is vital to practical scheduling policies. However, the optimal determination of safety time has no counterpart in deterministic scheduling. *Safe scheduling* departs from the dominant paradigm in stochastic scheduling by considering safety time explicitly.

In stochastic inventory theory there are two general ways to determine safety stocks—by meeting service-level constraints or by minimizing the expected total costs due to overstocking and understocking. We use analogous approaches in safe scheduling. To incorporate service-level constraints, we replace the deterministic definition of “on time” by a stochastic one. Define the *service level* for job  $j$  as  $\Pr\{C_j \leq d_j\}$ , the probability that job  $j$  completes by its due date. We sometimes denote this probability  $SL_j$ . Let  $b_j$  denote a given target for the service level. Then the form of a *service-level constraint* for job  $j$  is

$$SL_j = \Pr\{C_j \leq d_j\} \geq b_j.$$

We say that job  $j$  is *stochastically on time* if its service-level constraint is met; otherwise, the job is *stochastically tardy*. A complete sequence is called *stochastically feasible* if all jobs are stochastically on time. The use of service-level constraints is widely accepted in practice, especially when it is difficult to estimate the relevant costs, but it has seldom been exploited in the stochastic scheduling literature. An early exception was the observation by Banerjee [3] that the EDD sequence maximizes the minimum service level. (Equivalently, EDD minimizes the maximum probability that a job will be tardy.)

The alternative to imposing service-level constraints is to explicitly consider economic factors. If we can model the true economic costs of various outcomes with a total penalty function, we can then look for a schedule that minimizes the expected total penalty. Because the penalty function includes the cost of creating a buffer as well as the cost of failing to meet due dates, the minimization of the penalty function automatically dictates optimal safety. Unfortunately, the difficulty with this approach lies in the practical problems of acquiring good cost data, especially when some cost elements are subjective. When costs are hard to identify, we fall back on the service-level approach.

Both alternatives allow us to incorporate considerations of safety, but they do not specify the scheduling problem completely. Thus, there are two major formulations of the safe-scheduling problem. One formulation treats due dates (and possibly release dates) as given



FIGURE 2. Classification of safe-scheduling examples.

	Accept/reject decision	Due-date decision
Service-level constraints	$U$ -problem with service-level constraints	Due-date tightness and tardiness trade-off
Costs for all outcomes	$U$ -problem with three rewards	Optimizing earliness and tardiness penalties

and determines which jobs to accept, which to reject, and how to sequence the accepted jobs. We illustrate this formulation in the context of the  $U$ -problem. The other formulation treats due dates as decisions and adjusts these parameters in the process of minimizing expected total penalty. Typically, when due dates are decisions, we assume that all jobs will be processed, but we could accommodate the accept/reject decision as well. In either case, optimal safety time is a by-product of the analysis. We illustrate the second formulation in the context of the E/T-problem.

Considering the two approaches to acknowledging safety in conjunction with the two problem formulations, we obtain four combinations, which we discuss in the following subsections. Figure 2 depicts the main examples.

Our illustrations are representative of problems that could be addressed by ongoing research, but many other problems could be pursued from the lens of safe scheduling.

### 3.1. Service-Level Approach to the $U$ -Problem

Earlier, we presented the deterministic  $U$ -problem as one of postponing the jobs assigned to set  $R$  so that jobs in set  $A$  can be completed on time. Because jobs in  $R$  are late, they can be postponed indefinitely without altering the number of late jobs. Thus, we may equivalently consider jobs in  $R$  to be rejected, and Moore's algorithm can be viewed as a procedure for minimizing the number of rejected jobs. Similarly, we treat the stochastic  $U$ -problem as a problem of accepting or rejecting jobs. From the safe-scheduling perspective, the stochastic  $U$ -problem comes with service-level constraints and calls for minimizing  $|R|$  subject to stochastic feasibility of the accepted jobs. The general version of this problem is known to be NP-hard (Kise and Ibaraki [7]), and no computationally practical solution procedure has been developed.

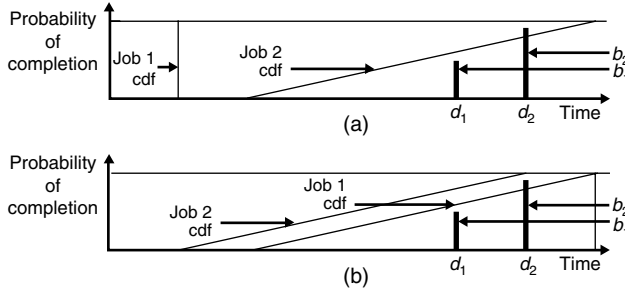
One complication that arises in the problem relates to the sequencing of the on-time jobs—those in set  $A$ . In Moore's algorithm, we can rely on EDD sequencing of set  $A$ , but the optimality of EDD does not generalize to the stochastic case, as the following example demonstrates.

Job $j$	1	2
$d_j$	5.0	6.0
$b_j$	0.5	0.9
$E(p_j)$	1	3.5

Job 1 has a deterministic processing time, but Job 2 has a processing time that follows a uniform distribution on the interval  $(1, 6)$ .

Figure 3(a) describes the situation for the EDD sequence. Two vertical segments with height 1, occurring at times 0 and 1, mark the start and finish of job 1. The uncertainty in  $C_2$  is represented by a linear cdf. Finally, the two service-level requirements are represented by vertical bars with heights  $b_j$  at the respective due dates. Clearly, Job 1 exceeds its service-level requirement because its cdf reaches a height of 1 prior to  $d_1$ . But Job 2 fails to meet its requirement because by  $d_2$  its cdf does not reach the height of  $b_2$ . Figure 3(b) demonstrates,

FIGURE 3. Graph for Example: (a) sequence 1-2, (b) sequence 2-1.



however, that if we interchange the jobs and sequence against EDD order, then both service-level constraints are met. The example shows that, contrary to the deterministic case, a stochastically feasible set of jobs is not necessarily feasible in EDD sequence.

Figure 3(a) is an example of a predictive Gantt chart. In a regular Gantt chart, jobs are always depicted as rectangles. Job 1 is depicted this way in Figure 3(a) because there is no uncertainty about its start and finish times. The height of its rectangle is equal to 1, and we can interpret the vertical line at time 1 as the cdf of  $C_1$ . The same job appears in Figure 3(b), but in that case its start time and finish time are uncertain and represented by cdfs. The area between the start cdf and the completion cdf of job 1 is the same in both figures and equals the expected processing time of the job. The horizontal line at the top of the figure can be interpreted as part of the cdf of the start time of the activity. As the figure demonstrates, a predictive Gantt chart shows the probability of completion as a function of time (because it involves cdfs) and can also be used to check whether particular service levels are met.

As we shall see, an important condition for solving this problem is stochastic ordering. The processing times of two jobs,  $j$  and  $k$ , are said to be *stochastically ordered* if  $\Pr\{p_j \leq t\} \geq \Pr\{p_k \leq t\}$  for any value of  $t$ . This does not mean that job  $j$  is always shorter than job  $k$ . Rather, for any duration  $t$ , the completion of  $j$  by this amount of time is more likely than the completion of  $k$  by this amount of time, and this relation holds for any duration  $t$ . In our example, we have a somewhat extreme case of stochastic ordering, because job 2 is always at least as long as job 1. Thus, our example illustrates that the stochastic infeasibility of EDD may occur even if processing times are stochastically ordered.

To make progress for our formulation of the  $U$ -problem, we need a procedure for determining whether a feasible sequence exists for any subset of accepted jobs. As it happens, a relatively simple procedure is available (van den Akker and Hoogeveen [14]). This *feasibility check* examines whether any job would satisfy its service-level constraint if scheduled last. (The probability distribution of the last job's completion time can be determined without knowing the full job sequence because the last completion time does not depend on the job sequence.) Any such job may be scheduled last and removed from further consideration. The procedure is then repeated for the remaining jobs. Because the procedure is constructive, it builds a sequence from back to front and yields a feasible sequence whenever the set of jobs is feasible.

Next, we imbed the feasibility check in the logic for accepting or rejecting jobs. If the jobs are stochastically ordered, we can use the logic of Moore's algorithm, with the feasibility check in place of EDD sequencing. The steps are as follows.

#### SEPT-Based Feasibility Check

1. Sequence the jobs by SEPT (ties may be broken arbitrarily) and place all jobs in the unresolved set (sets  $A$  and  $R$  are empty).

TABLE 5. Data for a five-job example.

	Job $j$	1	2	3	4	5
	$d_j$	7.8	7.5	17	20	12
State	$b_j$ (%)	90	60	50	80	60
1	$p_j$	2.6	3.5	3.8	4.5	6.4
2		2.8	3.9	4.4	5.5	6.6
3		3.2	4.1	5.6	6.5	7.4
4		3.4	4.5	6.2	7.5	7.6
	$E(p_j)$	3.0	4.0	5.0	6.0	7.0

2. Tentatively add the first unresolved job to  $A$  and apply the feasibility check. If the result is feasible, record the sequence and add the job to  $A$  permanently. Otherwise, add the job to  $R$ .

3. If the unresolved set is not empty, return to Step 2. Otherwise, stop. The last recorded sequence of the jobs in  $A$  is optimal.

To illustrate the application of the algorithm, we consider another five-job example, with service-level targets  $b_j$  as specified in Table 5.

When we consider the set  $\{1\}$ , we find that job 1 meets its target service level because it is certain to complete by its due date ( $d_1 = 7.8$ ), so in Step 2, we add job 1 to set  $A$ . When we consider the set  $\{1, 2\}$ , we find that job 1 cannot meet its target if it follows job 2, but job 2 meets its target if it follows job 1. Therefore, in Step 2, we add job 2 to  $A$ . Next, job 3 is feasible in the third position and is added to  $A$ . When we try to add job 4, it is not feasible in the fourth position, but the initial sequence 1-2-4-3 is feasible, so we add job 4 to  $A$ . Lastly, adding job 5 leads to infeasibility—no job can be feasibly scheduled in the last position. Therefore, job 5 is rejected, and the optimal solution is  $|R| = 1$ .

Note what happens if we use Moore’s algorithm, which relies on EDD sequencing of the jobs in set  $A$ . The EDD sequence is 2-1-5-3-4. jobs 2 and 1 are added to  $A$ , but job 5 cannot be feasibly added next, so it is assigned to set  $R$ . Then job 3 is added, but job 4 cannot be feasibly added after job 3, so job 4 is assigned to  $R$ , leaving only three jobs in  $A$ . As shown above, the optimal number of jobs in  $A$  is 4.

We are left with the question whether special cases exist in which we can rely on the EDD sequence for the stochastically on-time jobs. The answer is affirmative. For example, suppose that all jobs have the same service-level target—that is,  $b_j = b$ . In this case, a set of jobs is feasible if each one has a service level no lower than  $b$ . But as we mentioned earlier, the EDD sequence maximizes the minimum service level, so it will generate a sequence, if one exists, in which each service level meets or exceeds  $b$ . In other words, if a feasible sequence exists, the EDD sequence will be feasible.

We can also generalize the case of equal service levels. For any two jobs  $i$  and  $k$ , if  $d_i \leq d_k$  and  $b_i \geq b_k$ , then these parameters are said to be *agreeable*. When (1) every pair of jobs has agreeable due dates and service-level targets and (2) the processing times are stochastically ordered, the EDD sequence generates a feasible sequence if one exists. Thus, when those two conditions hold, Moore’s algorithm can be used to minimize the number of stochastically tardy jobs.

To summarize, the problem of minimizing the number of stochastically tardy jobs is NP-hard in general. If we know that the processing times are stochastically ordered, then we can find solutions with an algorithm based on a feasibility check, and if we also know that due dates and service-level targets are agreeable, then we can find solutions with Moore’s algorithm, which is slightly simpler. Using a safe-scheduling approach—in this case, making service levels part of the problem—we can at least find efficient solutions to special cases of a problem that is otherwise quite difficult to solve.

### 3.2. Economic Approach to the $U$ -Problem

The economic approach involves specifying the costs for various outcomes and then minimizing an objective corresponding to expected total penalty. If we can specify costs, then the minimization of total penalty may reflect reality better than a traditional summary measure such as  $U$ .

In the stochastic case, there is a conceptual difference between a job that is rejected by design and a job that is tardy by chance. That is, we may accept a job with the intention of completing it on time, but the stochastic nature of its processing time (and those of earlier jobs) may result in tardiness in spite of our intention. This structure leads us to a model that distinguishes three types of outcomes rather than two. For every job that is completed early or on time, we obtain a reward of  $R_E$ , a tardy job generates a reward of  $R_T < R_E$ , and the reward for rejecting a job is  $R_R$ . We require

$$R_E > R_R > R_T.$$

Here, our objective is to maximize the expected total reward. The assumption in the ordering of the three reward parameters is economically sound: If  $R_R$  were not strictly higher than  $R_T$ , we would gain nothing by rejecting a job—that is, we would want to process all jobs even if they are in  $R$  because the reward would be at least as good as rejecting them. Similarly, if  $R_R$  were not strictly lower than  $R_E$ , we would reject all jobs immediately. By subtracting  $R_R$  from all rewards, we change the total reward by a constant, but the optimal sequence does not change. Therefore, without loss of generality, we may assume that  $R_E > 0$ ,  $R_R = 0$ , and  $R_T < 0$ . After this adjustment, any optimal solution must have a nonnegative expected reward because rejecting all jobs would guarantee a total reward of zero.

To facilitate the incorporation of explicit rejection decisions into our model, we turn to the service-level approach and derive a constraint that all accepted jobs must satisfy. As introduced earlier,  $SL_j$  denotes the probability that job  $j$  is on time. Then the expected reward  $E(R_j)$  for job  $j$  when it is accepted becomes:

$$E(R_j) = R_E SL_j + R_T(1 - SL_j).$$

Note that this contribution is not positive unless

$$E(R_j) = R_E SL_j + R_T(1 - SL_j) > 0$$

or

$$SL_j > -R_T/(R_E - R_T).$$

The right-hand side of this inequality serves as a legitimate probability because  $R_T < 0$ . Furthermore, rejecting any job can only help reduce the tardiness of other jobs, so the optimal solution cannot call for accepting any job whose expected reward is negative. Therefore, if the inequality is violated for any accepted job at any sequence position, that sequence cannot be optimal. The inequality is necessary for each job, but not sufficient for optimality. In practice, these constraints are not likely to be tight except for jobs that are sequenced at the end of the schedule. In other words, early in the sequence it may even be suboptimal to accept a job whose service level barely satisfies the inequality. Although accepting such a job would generate a positive expected reward for the job itself, the consequence may be to reduce the service levels of several other jobs and ultimately lead to a net loss.

The economic approach to the  $U$ -problem is new, and no optimizing algorithm has yet been developed. Nevertheless, the problem can be attacked using the spreadsheet-based heuristic algorithm demonstrated earlier. Table 6 shows a spreadsheet layout and an initial solution for the five-job example introduced in the previous subsection with rewards of  $R_E = 5$  and  $R_T = -10$ . Once again, a good heuristic solution can be obtained with Evolutionary Solver, which generates an expected reward of 15 in this example.

TABLE 6. Spreadsheet solution to the example problem.

Example	Expected rewards with A/R decisions						
Data	Job $j$	1	2	3	4	5	$R_E$
Scenario	$d_j$	22	37	28	75	54	$R_T$
GG	$p_j$	10	8	12	15	20	5
GB		12	12	14	17	20	−10
BG		14	16	16	20	20	
BB		16	20	22	24	24	
	$E(p_j)$	13.0	14.0	16.0	19.0	21.0	
	Sequence	5	4	3	2	1	$E(\text{Reward})$
	Accept	1	1	1	0	1	−10.00
Processing times							
GG		20	15	12	0	10	
GB		20	17	14	0	12	
BG		20	20	16	0	14	
BB		24	24	22	0	16	
Completion times							
GG		20.0	35.0	47.0	47.0	57.0	
GB		20.0	37.0	51.0	51.0	63.0	
BG		20.0	40.0	56.0	56.0	70.0	
BB		24.0	48.0	70.0	70.0	86.0	
Rewards							Total
GG		5.0	5.0	−10.0	0.0	−10.0	−10.00
GB		5.0	5.0	−10.0	0.0	−10.0	−10.00
BG		5.0	5.0	−10.0	0.0	−10.0	−10.00
BB		5.0	5.0	−10.0	0.0	−10.0	−10.00

### 3.3. Due Dates as Decisions in the Tightness-Tardiness Trade-off

Due dates are considered given information in the basic model. However, some research (e.g., Baker and Bertrand [1]) has also explored models with due dates as decisions. This treatment is justified by the fact that due dates are sometimes negotiated with customers, so that the scheduling agent has some control over setting due dates. In addition, in multistage processes, due dates are often set internally by a control system, as a means of establishing progress guidelines while the jobs are in process. In what follows, we examine models that treat due dates as decisions.

In the first of these models, we are concerned about  $E(T)$  as a performance measure. When due dates are controllable, an inherent trade-off occurs. We can make tardiness small (even zero) by choosing the due dates to be sufficiently loose. On the other hand, tight due dates are desirable because they impose some discipline on the progress of due-date-oriented jobs. Therefore, we consider an objective function that combines due-date tightness and tardiness performance. More specifically, the objective function takes the form

$$f(S) = \sum d_j + \gamma \sum E(T_j),$$

where  $S$  represents the set of scheduling decision variables, here including the due dates. Instead of finding the optimal due dates directly, we can solve the problem by determining the optimal values of the service-level targets  $b_j$ . Once those optimal targets are found, they dictate the optimal due dates. Note that the objective function is additive: Each job makes a separate contribution to each of the two sums. Changing  $d_j$  affects these two contributions but not the contributions of any other job. Therefore, we can solve for the optimal due dates separately.

Because the effects of the due dates are separable, we can analyze the choice for job  $j$ . Suppose that we choose a due date of  $d_j$ , and then we observe a completion time of  $C_j$ . In retrospect, we can ask whether it would be desirable to have increased  $d_j$  by 1 initially. A unit increase in  $d_j$  would increase  $\sum d_j$  by 1 and would reduce tardiness if the job finished late—that is, if  $C_j > d_j$ . Thus, the net effect on the objective function would be  $1 - \gamma \Pr\{C_j > d_j\}$ . It follows that we should increase the due date as long as this expected incremental cost drops; that is, while

$$1 - \gamma(\Pr\{C_j > d_j\}) \leq 0.$$

Note that this inequality can never be satisfied unless  $\gamma > 1$ . For those values, we can equivalently write the required condition as increasing  $d_j$  while:

$$1 - \gamma(1 - \Pr\{C_j \leq d_j\}) \leq 0.$$

Therefore, we should set  $d_j$  equal to the smallest value for which this inequality fails and

$$\Pr\{C_j \leq d_j\} \geq (\gamma - 1)/\gamma.$$

This condition is a version of the *critical fractile rule* and justifies the following result.

**Theorem 6.** *Suppose the objective is to minimize  $\sum d_j + \gamma \sum E(T)$  for a given sequence. Then, if  $\gamma \leq 1$  we should set all due dates to 0. Otherwise, for job  $j$ , it is optimal to set  $d_j$  equal to the smallest value that satisfies the condition*

$$SL_j = \Pr\{C_j \leq d_j\} \geq (\gamma - 1)/\gamma.$$

As an example, consider the following five-job problem. The processing times are independent and each is drawn from a normal distribution with the mean and standard deviation shown in the table below. Take the parameter  $\gamma = 10$ .

Job $j$	1	2	3	4	5
$E(p_j)$	5	6	7	8	9
$\sigma_j$	1.8	1.0	0.3	0.8	0.2

In this instance, we have  $(\gamma - 1)/\gamma = 0.9$ , and the corresponding  $z$ -value in the normal table is  $z = 1.282$ . From this value we can calculate the due date for each of the five jobs, as shown in Table 7. We then calculate the mean tardiness for each of the jobs, using the algebra of normal distributions, and finally, the value of our objective function. In the example, the optimal value of this objective function is 112.851. In more detail,  $\sum d_j = 109.118$  whereas  $E(\sum C_j) = 99$ , and the difference is the combined safety time of 10.118. The expected tardiness cost is 3.733, for a total of 13.851 above  $E(\sum C_j)$ . Without safety time, however, the expected tardiness cost grows to 31.487 and the total cost to 130.487. Thus, a plan for 10.118 units of safety time leads to a net savings of 17.635 in the objective function.

Theorem 6 demonstrates that when due dates are decisions, they can be set to produce optimal service levels. That choice produces an optimal level of safety time for each job, although the specific level depends on the other jobs in the schedule.

Finally, whereas the optimal sequence to minimize  $\sum d_j + \gamma E(T)$  is not always easy to find, the SEPT sequence—which is optimal for the deterministic counterpart—is often a good heuristic. Indeed, if processing times are stochastically ordered, then it is possible to show that SEPT is optimal.

TABLE 7. Spreadsheet layout for the tightness-tardiness trade-off.

Example	$\sum d + \gamma \sum E(T)$	Sequence given						
<i>Normal case</i>								
Data								
	Job $j$	1	2	3	4	5	$\gamma$	10
	$E(p_j)$	5	6	7	8	9		
	$s_j$	1.8	1.0	0.3	0.8	0.2	$(\gamma - 1)/\gamma$	0.9
	Sequence	3	2	1	4	5		
	$E(p_j)$	7	6	5	8	9		
	$s_j$	0.3	1.0	1.8	0.8	0.2	$\sum d + \gamma \sum E(T)$	<div>112.851</div>
Calculations								
	Variance	0.090	1.000	3.240	0.640	0.040		
	Cumulative	0.090	1.090	4.330	4.970	5.010		
	Sq. root	0.300	1.044	2.081	2.229	2.238		
	Cum. mean	7	13	18	26	35		
	$z$ value 1.282	1.282	1.282	1.282	1.282			
	Normal pdf	0.175	0.175	0.175	0.175	0.175		
	Due date	7.385	14.338	20.668	28.858	37.870	109.118	
	Mean Tardiness	0.014	0.049	0.098	0.105	0.106	0.373	

### 3.4. Due Dates as Decisions in the Stochastic Makespan Problem

We next turn to a general economic approach in scheduling problems that treat due dates as decisions. We consider the stochastic version of the E/T-problem, and we begin with the special case of a common due date. The E/T-problem with a common due date arises when customers want their jobs completed at the same time. A more realistic application of a common due date arises when all jobs go to the same customer, feeding an assembly operation. In this case, the completion of the last job determines performance. The unit earliness cost  $\alpha$  and the unit tardiness cost  $\beta$  apply to the difference between the makespan ( $C_{\max}$ ) and the due date ( $d$ ). Thus, we let

$$f(S) = \alpha \max\{0, d - C_{\max}\} + \beta \max\{0, C_{\max} - d\}.$$

The deterministic version of this problem is trivial because the makespan is constant, and we can optimize  $f(S)$  by setting  $d = C_{\max}$ . However, in the stochastic case, the solution is nontrivial. For now, we assume no inserted idle time, and our objective is to minimize  $E[f(S)]$ . Then the job sequence is not at issue, and we can determine the optimal due date.

**Theorem 7.** Assume that all jobs are processed with no inserted idle time and that the objective is to minimize  $\alpha E[\max\{0, d - C_{\max}\}] + \beta E[\max\{0, C_{\max} - d\}]$ . Then the optimal due date corresponds to a service level of  $\beta/(\alpha + \beta)$ . That is, it is optimal to set  $d$  to the smallest value that satisfies the condition

$$\Pr\{C_{\max} \leq d\} \geq \beta/(\alpha + \beta).$$

This theorem, like the previous one, can be proved by applying critical fractile reasoning. In this case,  $\alpha$  represents the unit cost of overestimating the length of the makespan and  $\beta$  represents the unit cost of underestimating it. The resulting critical fractile, which appears in the theorem, is  $\beta/(\alpha + \beta)$ . A numerical illustration follows.

Consider a problem containing  $n = 5$  jobs with independent processing times. The processing time of each job follows a normal distribution, with mean and variance shown in the table.

Job $j$	1	2	3	4	5
$\mu_j$	5.0	7.0	6.0	4.0	8.0
$\sigma_j$	1.0	1.5	1.2	0.8	2.0

The unit penalties for earliness and tardiness are  $\alpha = 1$  and  $\beta = 4$ , respectively. The objective is to schedule the jobs so that the stochastic makespan is optimized.

In this example, we can model the properties of the makespan by drawing on the fact that the sum of five independent normal distributions follows a normal distribution, with mean and standard deviation denoted  $m$  and  $s$ , respectively

$$m = \mu_1 + \mu_2 + \cdots + \mu_5 = 30$$

and

$$s = (\sigma_1^2 + \sigma_2^2 + \cdots + \sigma_5^2)^{1/2} = 3.054.$$

The critical fractile is  $\beta/(\alpha + \beta) = 0.8$ , corresponding to a standard normal  $z$ -value of 0.8416, for which the optimal value of  $d = m + sz = 32.57$ . Thus, the asymmetry in  $\alpha$  and  $\beta$  leads to an optimal due date that is larger than the expected makespan. The difference is safety time.

The stochastic makespan model is highly specialized, however, and it is relatively simple because the job sequence is not at issue: Any job sequence generates the same makespan. Next, we address problems in which each job contributes separately to the E/T objective function and the job sequence matters.

### 3.5. Due Dates as Decisions in the Stochastic E/T-Problem

We now allow jobs to have different due dates as well as different unit costs. The objective function in the stochastic E/T-problem can be expressed as:

$$E[f(S)] = \sum_{j=1}^n [\alpha_j E(E_j) + \beta_j E(T_j)]$$

with due dates treated as decisions. The optimal choice of due dates is again determined by the critical fractile rule, as stated in the following result.

**Theorem 8.** *Assume all jobs are processed with no inserted idle time and the objective is to minimize the total expected E/T penalty. Then, for any given sequence, the optimal due date of job  $j$  corresponds to a service level of  $\beta_j/(\alpha_j + \beta_j)$ . That is, it is optimal to set  $d_j$  to the smallest value that satisfies the condition*

$$\Pr\{C_j \leq d_j\} \geq \beta_j/(\alpha_j + \beta_j).$$

For a given job sequence, we can apply Theorem 8 separately to all jobs and thereby minimize the total expected E/T penalty. However, it is algebraically challenging to apply the theorem unless we assume that the processing times follow a tractable probability distribution, such as the normal. Furthermore, it is important to remember that the theorem does not completely solve our problem: Although we can set the due dates optimally, there is no efficient procedure yet known for determining the optimal job sequence. Thus, the stochastic E/T-problem remains largely unsolved.

For some insight into a heuristic approach to solving this problem, assume that the processing times follow independent normal distributions. Let the processing time of the  $j$ th job



in sequence have mean  $\mu_j$ , and variance  $\sigma_j^2$ , and let the variance of the completion time of job  $j$  be  $s_j^2$ , where  $s_j^2$  depends on the sequence. Then, because the processing times are independent, we can calculate the mean ( $m_j$ ) and variance ( $s_j^2$ ) of the completion time for job  $j$  as follows:

$$m_j = \mu_1 + \mu_2 + \cdots + \mu_j,$$

$$s_j^2 = \sigma_1^2 + \sigma_2^2 + \cdots + \sigma_j^2.$$

We define the standard normal variate  $z_j = (d_j - m_j)/s_j$ , and we use an asterisk to denote optimal values. Thus, for example,  $z_j^* = (d_j^* - m_j)/s_j$  is the  $z_j$ -value that satisfies Theorem 8. In symbols,

$$z_j^* = \Phi^{-1}[\beta_j/(\alpha_j + \beta_j)],$$

where  $\Phi$  denotes the cdf of the standard normal distribution. That is, a safety time of ( $z_j^* s_j$ ) must yield a service level of  $\beta_j/(\alpha_j + \beta_j)$ , but note that  $m_j$  and  $s_j$  are not required to calculate  $z_j^*$ . Instead, we can calculate it directly from the normal cdf, setting the cumulative probability equal to the critical fractile. We then obtain:

$$d_j^* = E(C_j) + z_j^* s_j = m_j + z_j^* s_j.$$

Using the algebra of normal distributions, we can then write,

$$E[f(S)] = \sum_{j=1}^n \{(\alpha_j + \beta_j) s_j \varphi(z_j^*)\},$$

where we use  $\varphi$  to denote the standard normal density function. In the spirit of the deterministic counterpart, however, suppose that we also consider setting due dates equal to expected completion times:

$$d_j = E(C_j) = m_j.$$

(These due dates are equivalent to using safety times of zero.) Drawing on the algebra of normal distributions once more, we can show that the expected E/T cost for job  $j$  is  $(\alpha_j + \beta_j) s_j \varphi(0)$ . Therefore, the expected-value approach reduces to finding a sequence that minimizes

$$E[f(S)] = \sum_{j=1}^n (\alpha_j + \beta_j) s_j \varphi(0).$$

In the optimal schedule, the expected E/T cost associated with job  $j$ ,  $(\alpha_j + \beta_j) s_j \varphi(z_j^*)$ , cannot exceed  $(\alpha_j + \beta_j) s_j \varphi(0)$  because  $\varphi(z)$  is maximized at  $z = 0$ . The two ways of setting due dates may lead to different optimal sequences, and unless  $\alpha_j = \beta_j$  for all  $j$ , they lead to different objective function values. This analysis can become complicated, so we illustrate the concepts with a two-job example.

Consider a problem containing  $n = 2$  jobs, in which the processing times of the jobs follow independent normal distributions, and the objective is to minimize total expected E/T penalty.

Job $j$	1	2
$\mu_j$	10	10
$\sigma_j$	3	4
$\alpha_j$	3	9
$\beta_j$	27	36

Here,  $\alpha_1 + \beta_1 = 30$  and  $\alpha_2 + \beta_2 = 45$ . The optimal service levels are:

For job 1,  $\beta_1/(\alpha_1 + \beta_1) = 27/30 = 0.9$  (corresponding to  $z_1 = 1.282$ ),

For job 2,  $\beta_2/(\alpha_2 + \beta_2) = 36/45 = 0.8$  (corresponding to  $z_2 = 0.842$ ).

We solve for the best sequence by complete enumeration. For the sequence 1–2 we have  $s_1 = 3$  and  $s_2 = (3^2 + 4^2)^{1/2} = 5$ . Thus,

$$\begin{aligned} f(S) &= \sum_{j=1}^n (\alpha_j + \beta_j) s_j \varphi(z_j) = 30(3) \varphi(1.282) + 45(5) \varphi(0.842) \\ &= 30(3)(0.1755) + 45(5)(0.2800) = 78.79. \end{aligned}$$

For the sequence 2–1, we have  $s_2 = 4$  and  $s_1 = (3^2 + 4^2)^{1/2} = 5$ . Thus,

$$\begin{aligned} f(S) &= \sum_{j=1}^n (\alpha_j + \beta_j) s_j \varphi(z_j) = 45(4) \varphi(0.842) + 30(5) \varphi(1.282) \\ &= 45(4)(0.2800) + 30(5)(0.1755) = 76.72. \end{aligned}$$

Thus, the sequence 2–1 is optimal, and the corresponding due dates are as follows:

$$\begin{aligned} d_2 &= 10 + 4(0.842) = 13.368, \\ d_1 &= 20 + 5(1.282) = 26.410. \end{aligned}$$

Job 2 should complete around time 10 and job 1 around time 20. But if we follow the deterministic counterpart and set the due dates equal to the expected completion times of 10 and 20 for this sequence, the expected penalty would be 131.65. However, without safety times, the opposite sequence (1–2) would be better, with an expected penalty of 125.67, still about 70% above optimal. The implication is that the expected-value approach may not only lead to excessive penalty but also mislead the search for an optimal sequence.

Although there is no known optimizing algorithm short of complete enumeration, a good heuristic comes from a sorting rule. In particular, we can sort the jobs by nondecreasing ratio of  $\sigma_j^2/(\alpha_j + \beta_j)\varphi(z_j^*)$ , with ties broken in favor of the smallest  $\sigma_j$ . This heuristic is often optimal for small  $n$  (Soroush [12]).

For large  $n$ , this heuristic is asymptotically optimal. A heuristic is *asymptotically optimal* if, as  $n$  grows large, the relative difference between the heuristic solution and the optimum becomes negligible. More formally, let  $f(S^*)$  denote the objective function value based on the optimal sequence,  $S^*$ , and let  $f(S^H)$  be the value associated with a heuristic. We say that the heuristic is asymptotically optimal if, in the limit as  $n \rightarrow \infty$ , we have

$$[f(S^*) - f(S^H)]/f(S^*) \rightarrow 0.$$

Furthermore, for independent processing times, the ratio heuristic is asymptotically optimal even if the jobs are not distributed normally. This result follows from the fact that for large  $n$  the central limit theorem holds, and nearly all completion times are normally distributed. Among all possible sorting rules, only those consistent with this heuristic rule are asymptotically optimal. The tie breaker is not necessary for asymptotic optimality but it is useful because for the normal distribution, if the heuristic rule is consistent with nondecreasing  $\sigma_j$ , then it is known to be optimal (Portougal and Trietsch [9]).

Because asymptotic optimality does not require normal processing times, the heuristic sorting rule is effective for any processing-time distribution. Furthermore, we can use the sorting heuristic to find an initial seed, and then perform neighborhood searches on the first few jobs (say 5 to 10) to see if an even better sequence can be found. For subsequent jobs, we can rely on the asymptotic optimality of the heuristic. We can use this sorting rule as a crude heuristic even when jobs are not statistically independent. To estimate job parameters for that purpose, we use their marginal distributions. After the sequence is determined, we can set the due dates using the critical fractile rule, which does not require statistical independence or normality. Although the use of marginal distributions is not theoretically correct, this method can at least generate a reasonable seed for a neighborhood search.

## 4. The Benefit of Inserted Idle Time

Consider the stochastic E/T-problem with a common due date, as it is a special case that can teach us something about the more general case with distinct due dates. The deterministic counterpart of this problem exhibits three general properties (Baker and Scudder [2]). These are: (1) inserted idle time provides no benefit, (2) a V-shaped schedule (processing times are first nonincreasing, then nondecreasing) is optimal, and (3) one job completes at the due date. In the stochastic case, of course, we would not expect the last condition to hold, but we might hope the other properties apply. However, as we might guess from examples with the normal distribution, V-shaped schedules may not be optimal because they do not account for variance. The remaining question involves inserted idle time. Unfortunately, we may encounter limited success relying on this feature of the deterministic counterpart to solve the stochastic problem. To illustrate, we consider an example with a common due date and symmetric earliness and tardiness costs that are the same for all jobs.

Job $j$	1	2	3
$E(p_j)$	3.4	1	1
$d_j$	10	10	10
$\alpha_j$	1	1	1
$\beta_j$	1	1	1

The processing times depend on which of two states of nature occur, as described in the following table.

State	Job $j$	1	2	3	Probability
$S_1$	$p_j$	1	1	1	0.2
$S_2$	$p_j$	4	1	1	0.8

Note that  $p_1$  is a random variable, but the other two jobs have deterministic processing times. In the deterministic counterpart, job 1 comes first, and the other jobs follow in either order. The sequence begins at time 5.6, so that the second job completes at time 10, and the total E/T penalty is 2.

Now suppose we implement the sequence 1–2–3 in the stochastic case. If we start job 1 at time 5.6, the expected total penalty is 3.52. If we explore other starting times, we find that starting job 1 at time 5.0 leads to an expected total penalty of 3.4, which is the best objective for this sequence. If we schedule job 1 last, the best we can do is to start the schedule at time 8, leading to an expected total penalty of 4.4.

Next, we explore the possibility of inserting idle time in the sequence 1–2–3. Suppose we start the schedule at time 5 but prevent the second job from starting earlier than time 9. In other words, when job 1 completes at time 6 (which occurs with probability 0.2), we allow idleness until time 9, when job 2 starts. This schedule achieves an expected total penalty of 2.6, which is better than we could achieve with no inserted idle time.

This example reveals a complicating factor in stochastic problems with E/T criteria: It may be helpful to allow inserted idle time between jobs, even though (for the case of a common due date) such idle time would not be beneficial in the deterministic counterpart. Thus, we must pay attention to the general case in which inserted idle time is permitted.

### 4.1. Setting Release Dates with Known Due Dates

As the previous example shows, inserted idle time can be beneficial in the stochastic case. The constraint on the start time for the second job is essentially a *release date*,  $r_j$ , for job  $j$ . If the machine is available before  $r_j$ , the machine must wait to start job  $j$ ; but if the machine becomes free after  $r_j$ , the job can start immediately.

It is not always necessary to assign an explicit release date to each job. We may start a search for optimal release dates under the assumption that each job has its own release date, but it is ultimately sufficient to describe a schedule by specifying only release dates that have a positive probability of actually delaying a job. We refer to such release dates as *active*. A release date that is not active is redundant because it never causes a machine to wait.

Release dates define blocks. A *block* is a sequence of jobs processed without delay. If no release date is specified for a job, it is in the same block as the preceding job. (The only exception would be at the start: If no release date is specified for the first job, then processing starts at time 0.) In the stochastic case, adjacent blocks may be processed with or without a gap between them, but the expected size of the gap is positive. In the optimal schedule for the three-job example above, we place job 1 first in sequence and take  $r_2 = 9$ . Job 1 thus belongs to block 1, whereas the other two jobs make up block 2 and the expected gap is  $0.2 \times 3 = 0.6$ .

Suppose we are given a set of jobs with distinct due dates and E/T costs, and suppose further that the job sequence is given. The task then is to set release dates that minimize the total expected E/T penalty. It is possible to show that the total expected E/T penalty is a convex function of the release dates. Essentially, we need to search for the best combination of release dates to minimize this total expected penalty.

When we use a sample-based approach, we can find the best combination of release dates by a numerical search, because the problem is convex and thus not difficult in practice. Our model determines optimal release dates for a given sequence, but we still do not have an efficient algorithm for finding the best sequence. For the time being, a heuristic procedure, such as a neighborhood search algorithm, represents the state of the art for finding the optimal sequence.

In this section, we have examined the assumption of no inserted idle time and discovered that it may be restrictive. Optimal solutions to the stochastic E/T-problem may require that idle time be inserted in the schedule, or equivalently, that explicit release dates be set for some of the jobs. At this stage, we have not pursued the full implications of this observation, leaving that task to future research.

## 5. Summary and Conclusions

In this tutorial, we presented the basic single-machine model, which is central to most of scheduling theory, and we gave an overview of traditional extensions to stochastic versions of the basic model. We discussed in detail the shortcomings of using the deterministic counterpart as a device for attacking stochastic problems. First, it ignores the Jensen gap, which is the source of bias in predicted performance. Second, it ignores the practical value of safety time. Although deterministic models may be very useful as sequencing heuristics, they tend to be inadequate unless safety time is addressed correctly.

In practical applications, the bias due to the Jensen gap often has serious consequences. Because of the optimistic bias in deterministic counterparts, the natural way to use a deterministic model for a stochastic situation is to build hidden buffers into each job's estimated processing time. Such hidden safety time is ultimately wasteful. For this reason, traditional stochastic-scheduling models have not been suited to practical applications, even though real problems are often stochastic.

Safe scheduling calls for recognition of safety time, often expressed in service levels or critical-fractile results. We introduced safe scheduling and discussed four general types of models that arise in safe scheduling. Using an analogy to stochastic inventory theory, we identified two approaches to determining safety time: meeting service-level constraints or minimizing the expected value of total economic cost. We considered each of these approaches with two options for achieving safety: (1) Rejecting jobs whose inclusion could undermine the safety of other jobs or (2) accepting all jobs but assigning due dates in an optimal fashion. The combination of the economic approach and due-date setting can be viewed as the

stochastic counterpart of deterministic models with E/T penalties. In our coverage of this combination, we assumed no inserted idle time, but we discussed the fact that this assumption may be restrictive. An open area for research appears to be the analysis of release dates to control inserted idle time.

We foresee the themes illustrated in these types of models as giving rise to new opportunities for research and application in stochastic scheduling. Our main conclusion is this: If we wish to fully utilize our historic investment in scheduling theory, we must pay more attention to safe-scheduling formulations.

## References

- [1] K. R. Baker and J. W. M. Bertrand. A comparison of due-date selection rules. *AIIE Transactions* 13:123–131, 1981.
- [2] K. R. Baker and G. D. Scudder. Sequencing with earliness and tardiness penalties: A survey. *Operations Research* 38:22–36, 1990.
- [3] B. P. Banerjee. Single facility sequencing with random execution times. *Operations Research* 13:358–364, 1965.
- [4] T. B. Crabill and W. L. Maxwell. Single-machine sequencing with random processing times and random due dates. *Naval Research Logistics Quarterly* 16:549–555, 1969.
- [5] J. R. Jackson. Scheduling a production line to minimize maximum tardiness. Research Report 43, Management Sciences Research Project, University of California at Los Angeles, Los Angeles, CA, 1955.
- [6] S. M. Johnson. Optimal two- and three-stage production schedules with setup times included. *Naval Research Logistics Quarterly* 1:61–68, 1954.
- [7] H. Kise and T. Ibaraki. On Balut's algorithm and NP-completeness for a chance constrained scheduling problem. *Management Science* 29:384–388, 1983.
- [8] J. M. Moore. An  $n$ -job, one-machine sequencing algorithm for minimizing the number of late jobs. *Management Science* 15:102–109, 1968.
- [9] V. Portougal and D. Trietsch. Setting due dates in a stochastic single-machine environment. *Computers & Operations Research* 33:1681–1694, 2006.
- [10] M. H. Rothkopf. Scheduling with random service times. *Management Science* 12:707–713, 1966.
- [11] W. E. Smith. Various optimizers for single-stage production. *Naval Research Logistics Quarterly* 3:59–66, 1956.
- [12] H. M. Soroush. Sequencing and due-date determination in the stochastic single-machine problem with earliness and tardiness costs. *European Journal of Operations Research* 113:450–468, 1999.
- [13] W. Szwarc, A. Grosso, and F. Della Croce. Algorithmic paradoxes of the single-machine total tardiness problem. *Journal of Scheduling* 4:93–104, 2001.
- [14] J. M. van den Akker and J. A. Hoogeveen. Minimizing the number of late jobs in case of stochastic processing times with minimum success probabilities. *Journal of Scheduling*. Forthcoming.

# Community-Based Operations Research

**Michael P. Johnson**

Department of Public Policy and Public Affairs, University of Massachusetts Boston, Boston, Massachusetts 02125, michael.johnson@umb.edu

**Karen Smilowitz**

Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, Illinois 60208, ksmilowitz@northwestern.edu

**Abstract** *Community-based operations research* is defined as the collection of analytical methods applied to problem domains in which interests of underrepresented, underserved, or vulnerable populations in localized jurisdictions, formal or informal, receive special emphasis, and for which solutions to problems of core concern for daily living must be identified and implemented so as to jointly optimize economic efficiency, social equity, and administrative burdens. As such, it represents a specific domain within public-sector OR. Community-based operations research (OR) problems tend to be “messy” and highly dependent on political and social considerations. Nevertheless, solution of these problems is essential to the continued health and well-being of individuals, families, communities, and entire regions. This tutorial emphasizes current research in a variety of application areas. We identify a tension between problems that reflect unique characteristics of local communities and those that represent more general characteristics that bridge diverse communities. We propose principles for bridging the gap between quantitative model- and method-based approaches typically associated with private-sector problems, and qualitative policy and process-oriented approaches typically associated with public-sector problems. We examine two research applications in detail: food security and affordable housing. In so doing we emphasize the commonality of problem attributes and the diversity of modeling and solution methods.

**Keywords** public sector; policy modeling; urban affairs; multiple-objective optimization; equity; non-profit operations research

---

## 1. What is Community-Based OR?

### 1.1. Introduction

Many public-sector problems in the United States of interest to researchers and practitioners have origins and solutions with a strong local or community flavor and reflect concern with individual life outcomes. Such problems include poverty, food security, urban education, criminal justice, and human services. Indicators of the severity of these problems are numerous. The U.S. poverty rate, about 12.6% in 2005, has not decreased significantly over the past three years despite growth in multiple economic indicators. Moreover, the poverty rate for African Americans and Hispanics remains about three times that of non-Hispanic whites (U.S. Census Bureau [86]). Performance of public and private elementary and secondary school students on a national measure of academic achievement has shown only modest improvements between 1999 and 2004, with non-Hispanic white students two to three times more likely to score at the highest levels (National Center for Educational Statistics [71]). Unemployment rates for African American males without college education have increased dramatically over the past 15 years, with rates two to three times that of whites; incarceration rates for African Americans in their late 20s are about six times that of whites

(Western [95], Bureau of Justice Statistics [17]). The number of U.S. households with at least moderate housing burdens (at least 30% of income) rose from 31 to 34 million between 2001 and 2004, whereas average commute times increased at greater rates for lower-income families than for higher-income families (Joint Center for Housing Studies [47]).

Research on social issues such as those described above has traditionally been descriptive in nature, and concentrated within the social sciences, urban planning, and related disciplines; they have tended to receive somewhat less attention from disciplines such as operations research/management science (OR/MS) whose models and methods tend to be more prescriptive. We assert that an adaptation of previously identified domains of OR/MS, which we denote as *community-based operations research*, is well positioned to provide models and analytic methods that can provide guidance to individuals, government, and non-governmental organizations that seek to address these problems. In this tutorial, we describe the characteristics of problems that fall within the scope of community-based OR, develop a taxonomy of problem areas, provide a review of related OR/MS research literature, and explore two case studies of community-based OR applications. Finally, we identify some next steps for research and practice in this area.

The scope of this tutorial is largely limited to the U.S., as American political and social values, and demographic and economic characteristics may differ from other developed countries in certain ways relevant to community-based OR. We focus on the last 30 or so years of research and applications, the scope of time within which “public-sector operations research” has become a well-defined domain. We define OR/MS broadly to include prescriptive decision models as well as descriptive models that are directly applicable to decision modeling. Evidence to support the presence of community-based OR comes primarily from refereed journal articles, published books, and conference proceedings, although working papers are cited if they provide primary evidence of recent research activity in a given domain.

## 1.2. Defining Community-Based OR

Fundamentally, community-based OR addresses public-sector problems, that is, problems in which the primary outcome measure to be optimized is not a direct representation—or proxy for—shareholder benefit and in which the outputs are subject to public scrutiny (see, e.g., Pollock and Maltz [75], p. 6). Community-based OR is a subfield of public-sector OR (see e.g., Pollock et al. [76], Larson and Odoni [60]), which emphasizes most strongly the needs and concerns of disadvantaged human stakeholders in well-defined neighborhoods. Within these neighborhoods, localized characteristics vary over space and exert a strong influence over relevant analytic models and policy or operational prescriptions. There are three important implications of this definition that distinguish community-based OR from other areas of public-sector OR.

First, our focus on human stakeholders implies a fundamental interest in human versus physical resources, as might be represented by decision models for natural resources management, energy policy, and—to a lesser extent—infrastructure design. Second, our focus on disadvantaged, underserved, or vulnerable populations recognizes that these groups may have distinct social or political preferences, are less able to use political influence to adapt public policies to their own needs, and may suffer stigma that may impede access to resources, expertise and tools associated with state-of-the art decision models or policy interventions. Third, our focus on “place” implies that models should account for community-level characteristics, like socio-economic status, that are salient to decision models and which vary over space in systematic ways. For example, suppose a public-library-branch location model does not distinguish between demands originating from neighborhoods that differ according to socio-economic status. Then a poor and nonpoor neighborhood with equal levels of demand may be treated equally by the model, even if relative travel costs are higher and marginal benefits from access to recreational and educational resources greater for residents of poor neighborhoods as compared to nonpoor neighborhoods.

Community-based OR has its roots in a classic article by Ackoff [2] describing community-engaged problem solving in a distressed, mostly African American community in Philadelphia. This amalgam of analysis, action, and advocacy received the greatest attention in the United Kingdom during the 1980s and 1990s as *community operational research* (Jackson [37], Parry and Mingers [73], Taket and White [83]), although some attention has been given to this sort of OR practice and theory in the U.S. (e.g., Bajgier et al. [5]). An important goal of this tutorial is to respond to Parry and Mingers' statement that "very little has been published on Community OR, and as a result it is unclear how much has been achieved" ([73], p. 580). These authors demonstrated that much of the work in community OR appeared to consist of student projects and community outreach that might be classified now as "capacity-building," rather than applications based on analytic models intended to provide specific policy and operational guidance to decision makers in a way that extends existing theory and methods. Most of the applications of community-based OR described in this tutorial have occurred since Parry and Mingers' [73] article, and address the methodological concerns noted above.

Certain public-sector problems are not amenable to community-based OR. As mentioned previously, many social problems, such as criminal offending, low-quality housing, or inadequate access to retail food outlets exhibit symptoms that vary across neighborhoods. These problems may also have important regional or national characteristics that are aggregates of individual outcomes, for example high incarceration rates, residential segregation, or adverse health outcomes related to food insecurity. In turn, these high-level characteristics may be associated with regional or national-level policy failures, such as ineffective national drug enforcement policy, inadequate funding for affordable housing, or insufficient incentives for grocery stores to locate in underserved communities. An important area of public-sector OR, referred to by Kaplan [52] as *policy modeling*, uses stylized models inspired by problems of a local nature that may generate important insights regarding regional or national-level policy design. Policy modeling has yielded novel and influential insights in drug policy (see e.g., Rydell et al. [79]), crime policy (see e.g., Blumstein et al. [12]), and public health (see e.g., Kaplan and Merson [55]). Community-based OR, although quite distinct from policy modeling, can complement policy modeling by generating solutions associated with direct and rapid improvements in individual and neighborhood-level outcomes.

We may summarize the defining characteristics of community-based OR problems as follows. They tend to have *multiple stakeholders and multiple decision makers* (Gass [26], Bardach [8]). We enumerate typical stakeholder and/or decision-maker groups as follows. Donors are government, nonprofit, or for-profit organizations whose direct or in-kind contributions support service provision. Clients are individuals, families, or organizations who benefit directly from service provision. Nonclient residents are individuals, families, or organizations who benefit indirectly or not at all from service provision but who may nevertheless pay for it through taxes. Service providers are government or non-profit organizations who design, implement, and manage service provision strategies. Local government enforces laws, administrative rules, and codes that define the legality of various initiatives. As we have emphasized, key stakeholders and decision makers are *localized* and often economically or socially *disadvantaged*. Therefore, trade-offs between efficiency, effectiveness, and equity are essential. In contrast to other private- and public-sector OR models, limiting focus to a single decision maker, stakeholder, or objective function type may obscure important aspects of the problem at hand.

As for United Kingdom-style community OR, successful community-based OR models and applications require substantial *stakeholder participation* in problem definition, solution, and implementation (Bajgier et al. [5], Taket and White [83]). This is incompatible with the conventional "consultant" view of OR modeling in which a dispassionate expert becomes immersed in a problem domain, formulates and solves an appropriate analytical model, and presents a range of recommendations to decision makers.



*Accountability*, a traditional focus of public administration generally (Heinrich [35]; see also Gates [27]) is especially important for community-based OR for determining the social impact of model solutions. Many important community-based services, targeted as they are on disadvantaged populations, generate relatively low levels of user fees and rely disproportionately on support from nongovernmental organizations. In turn, these donors may want service providers to demonstrate that they have achieved significant client and system outcomes—but these are difficult to measure and communicate, as compared to process outputs. The implications of this criterion are twofold: There is an increased need for researchers to design effectiveness measures that can be easily implemented in community OR models, and there is an increased need for practitioners to work closely with funders to justify the support received.

Finally, there is a general tension between *uniqueness* and *generalizability* that affects how community-based OR models and applications are viewed by disciplines that might take an interest in them. The greater the programmatic and spatial specificity of a given application and the focus given to community engagement, the greater is the resemblance of community-based OR to domains at the intersection of community planning, community organizing, and social work. As Kaplan [52] observes, this view of community-based OR may have led Ackoff [3] to become disenchanted with the prospects of “traditional” OR as a vehicle for making significant changes in society. The greater the generality of the model and the application, the greater is the resemblance of community-based OR to decision modeling applications whose contribution is primarily technical and methodological, or those, like policy modeling, that take a regional or national focus, decreasing the likelihood of direct, relatively rapid benefits to community stakeholders. Our belief is that, all else equal, the long-term impact of community-based OR is likely to be greatest if emphasis is placed on models that provide specific, theory-based guidance to local decision makers that can be easily replicated in different application contexts.

### 1.3. Sample Application: Public-School System Design

An example of an application that captures many of the characteristics of community-based OR described above is that of an urban public-school district facing declining enrollment, increasingly rigorous educational quality targets and revenue shortfalls that must simultaneously decide on a set of schools to close and to combine, academic programs to relocate, and students to reassign (see e.g., Mar Molinero [64], and Johnson [41]). Table 1 demonstrates the various dimensions along which community-based OR can help decision makers and stakeholders collaborate to generate specific recommendations for policy and operational guidance.

Results indicate that this problem is rich, in terms of opportunities for community engagement, mathematical modeling complexity, data requirements, and multi-stakeholder decision support. A traditional OR approach to this application might emphasize sophisticated, high-quality solution algorithms for a somewhat stylized representation of the underlying problem, perhaps using simulated data, and a focus on efficiency and possibly effectiveness measures. A community-based OR approach might instead result in a model that captures the local environment more fully, and incorporate equity and effectiveness explicitly, thus implying greater difficulty in optimal solution procedures. However, this approach might also result in heuristic solution algorithms and strategies for model formulation and policy implementation that emphasize qualitative methods more typical of UK-style community OR.

### 1.4. A Taxonomy of Community-Based OR Applications

There is no concise, standard classification of public-sector service areas known to us (a comprehensive list of service areas is available on the U.S. Blue Pages, <http://www.usbluepages.gov>). For the purposes of this tutorial, we classify community-based OR application areas

TABLE 1. Example community-based OR application: Urban public-school closings.

Attribute	Description
Localized focus	A school-closing policy may differentially affect low-performing schools, and disadvantaged neighborhoods, as compared to schools and neighborhoods overall. Therefore, generic modeling constructs may not capture the most challenging aspects of urban public-education policy.
Multiple conflicting objectives	Efficiency: direct dollar savings in fixed and variable costs Effectiveness: changes in student educational performance Equity: changes in average school travel times across neighborhoods; changes in levels of racial, ethnic or class segregation across and within schools.
Multiple stakeholders	Donors: local school district, federal government, local foundations. Clients: families whose children attend public schools. Nonclient residents: households without school-age children, or whose children attend nonpublic schools. Service provider: local board of education.
Role of disadvantage	Racial and ethnic minorities, who tend to be economically disadvantaged and segregated, may constitute a majority of students enrolled in public schools, although not necessarily a majority of the voting population, or cadre of professional analysts, funders, or political leaders.
Accountability	Cost savings are a direct consequence of school closings; improved educational outcomes are not. Educational outcomes may even worsen over the short term as the system re-equilibrates. School-closing decisions could be based, in part, on conventional measures such as standardized test scores—or, alternatively, on more sophisticated measures that identify the “value added” by schools. But specifying the latter might be controversial and expensive.
Lack of resources	Local boards of education may have no analysts with experience in OR/MS models and methods, and few hardware or software resources to solve the challenging models that arise in developing school-closing strategies.
Uniqueness versus generalizeability	Too much focus on local attributes shifts emphasis to politics, community organizing, and educational administration rather than a more generic model that can incorporate issues relevant to many different cities.

*Note.* Problem description: Johnson [41].

in four broad categories: human services, community development, public health and safety, and nonprofit management. *Human services* consist of services to senior citizens, humanitarian (e.g., post-disaster) logistics, public libraries and literacy, public education, and family supportive services. Family supportive services include e.g., foster care, income-based benefits such as food stamps and public assistance, and need-based benefits such as mental health/mental retardation, drug/alcohol treatment, and homeless services.

*Community development* consists of housing, community/urban planning, and transportation. In turn, housing can be classified as low- and moderate-income housing, often provided through government subsidies, and affordable, mixed-income, and workforce housing, often provided through zoning ordinances or private initiative as well as direct government support. Community and urban planning can address conventional economic development of distressed or isolated communities as well as pre-disaster planning and post-disaster reconstruction.

*Public health and safety* addresses health care, criminal justice, emergency services, hazardous/undesirable facility location, and the correlation of chronic diseases and individual deprivation or social externalities, such as food insecurity or proximity to environmental

hazards. Finally, *nonprofit management* addresses general issues in management of community-based or community-oriented service providers. In §2 we will review selected research literature in these broad application areas.

Table 2 displays a variety of community-based OR application areas according to multiple attributes such as geographic/temporal scope, performance/outcome metrics, and modeling/computational complexity. If there are multiple examples of published OR applications in a particular category, we emphasize the application we believe has the strongest community orientation.

Note that community-based OR problems can be operational, tactical, or strategic in nature; that analytical methods from logistics and multicriteria decision-analysis/decision theory tend to dominate, and that there are multiple outcome measures of interest, some of which may require knowledge of economics to quantify (e.g., net social benefit), others of which may require close examination of stakeholder values (e.g., equity).

### 1.5. What is Community-Based OR's Profile Within the Profession?

We conclude this introductory section by assessing the visibility of community-based OR within the OR/MS community. Community-based OR appears to have a low profile within top-tier disciplinary OR/MS journals. A review of articles published between 2002 and 2007 whose topics appear consistent with our definition of community-based OR yields four papers in *Operations Research* out of 401,<sup>1</sup> or 1%, and no papers in *Management Science*, *Transportation Science*, or *Information Systems Review*.

Community-based OR also appears to have a low profile within academic degree programs that intersect OR/MS. A scan of curricula of top schools, according to the 2007 rankings of *U.S. News and World Report* [90, 91], indicates that within the top 25 undergraduate industrial engineering/operations research programs, none appears to offer courses that address community-based OR; within the top 10 graduate industrial engineering/operations research programs, only one offers one or more courses whose content includes community-based OR. Within the top 25 undergraduate business programs, only one appears to offer one or more courses that address community-based OR; a similar result holds for the top 25 graduate programs in business. Finally, within the top 25 graduate programs in public affairs, only three offer courses with significant OR/MS content, and of these only one school's OR/MS course offerings appear to address community-based OR.

We acknowledge that much important work in OR/MS that intersects community-based OR is done outside of business schools, departments of industrial engineering and remains unpublished, or is published in non-OR/MS flagship journals. For example, pro bono OR consulting and applications, as advocated by Woolsey [97] and McCardle [66], often address problems of interest to public-sector organizations. However, we believe that a greater emphasis on community-based OR in education, research, and practice may increase the professional returns to those who work in this area. We make specific suggestions in this regard in the final section.

## 2. Literature Review

In the years since the Parry and Mingers (1991) survey of community OR, a number of papers have appeared that address their concerns. In the review that follows, we focus on studies in OR/MS and related fields that address analytic methods, public or nonprofit management, localized needs, equity concerns, or disadvantaged/underserved populations with a bias toward model-based prescriptions.

<sup>1</sup> Three of these papers appeared in a special 50th anniversary edition of *Operations Research* comprised of specially commissioned retrospectives.



TABLE 2. Continued

Application areas	Attributes					
	Example application(s)	Entity	Geographic/temporal scope	Performance/outcome metrics	Relevant methods	Modeling/computational complexity
Criminal justice	Location of community corrections centers; facilitating prisoner re-entry into communities	Government, NGO	Region, neighborhood; tactical	Net social benefit; equity	Location-allocation; forecasting	Moderate
Urban and regional development	Post-disaster reconstruction; design of community redevelopment initiatives; site selection and parcel acquisition	Government, NGO, public-private partnerships	Region, neighborhood; strategic/tactical	Net social benefit; equity; sustainability; spatial desirability	Location-allocation; districting; project selection; multi-criteria decision models; decision analysis	Moderate
Public health	Location/service design of health centers; blood distribution; design of public health initiatives	Government, NGO	Region, neighborhood, individual; strategic/tactical/operational	Net social benefit, equity, lives saved	Location-allocation; vehicle routing; inventory theory; stochastic processes; decision theory	High
Public libraries/literacy	Location/service design	Government	Region, neighborhood; strategic/tactical/operational	Social cost, physical access	Location-allocation, scheduling	Moderate
Undesirable facility location	Waste dumps, power plants, human services	Government, NGO	Region, neighborhood; strategic/tactical	Net social benefit, equity	Location-allocation, multi-criteria decision models	Moderate
Emergency services	Location of fire, police and EMS stations; dispatching, routing and staffing	Government	Regional, neighborhood; strategic/tactical/operational	Total cost, lives saved, crime averted, equity	Location-allocation, queuing theory, decision theory; vehicle routing	High

\*NGO = Nongovernmental organizations, e.g., community development organizations, social service agencies, churches.

## 2.1. Analytic Methods

**2.1.1. Quantitative Methods.** We know of no conventional quantitative modeling methods from OR/MS that cannot be applied to community-based OR problems. We draw attention to Erlenkotter's [23] examination of an extension of the uncapacitated fixed-charge facility location models to account for price-sensitive demands. He shows that a conventional private-sector objective of profit maximization is equivalent to marginal-cost pricing, whereas a public-sector objective of total social benefit maximization yields revenues that do not cover total costs. A "quasi-public" variant of this problem ensures nonnegative profits. This analysis reinforces our emphasis on community-based OR models that attempt to jointly optimize measures of equity and social welfare. The latter objective, moreover, is generally not equivalent to profit maximization.

**2.1.2. Qualitative Methods.** Qualitative methods provide a valuable complement to traditional approaches. UK-style community OR draws heavily from methods such as "soft systems methodology" (see e.g., Checkland [19]) that focus on systems analysis and qualitative methods for learning *about* the problem. Community-based OR, in addition, accommodates methods such as "value-focused thinking" (Keeney [56]) that assists modelers in formulating decision problems that are closely aligned with stakeholder values and amenable to analytical methods that yield specific prescriptions.

## 2.2. Human Services

**2.2.1. Senior Services.** Bartholdi et al. [9] develop a heuristic vehicle-routing strategy for home-delivered meals (HDM, i.e., meals on wheels) that can be easily implemented by the resource-constrained organization using earlier work on space-filling curves. Wong and Meyer [96] use geographic information systems (GIS) to locate HDM kitchens and design delivery routes to minimize total travel distance. Gorr et al. [28] develop an interactive, optimization-based spatial decision support system (SDSS) for HDM kitchen location, catchment area design, and vehicle routing for the needs of nonprofit managers seeking incremental or dramatic changes to service strategies. Johnson et al. [46] design and implement hierarchical facility location models to locate facilities that provide daytime congregate services to senior citizens that minimize distance-based costs and maximize utility using current local data on senior centers and demands for services.

**2.2.2. Humanitarian Logistics.** Haghani and Oh [34] consider operational transportation problems for a fixed distribution network and develop a model of the distribution process. Jia et al. [38] introduce network design models for large-scale emergency medical service in response to terrorist attacks. Balcik and Beamon [7] study distribution network design for relief chains managed by nongovernmental organizations. Campbell et al. [18] explore various objective functions for the local distribution of supplies after a disaster. To ensure equity and efficiency, two objectives are considered: minimizing the maximum arrival of supplies and minimizing the average arrival of supplies. Beamon and Kotleba [10] design a stochastic inventory control model that determines optimal order quantities and reorder points for a long-term emergency relief response.

**2.2.3. Public Libraries and Literacy.** Mandell [63] presents multiple models of equity and effectiveness, and applies them to the problem of book acquisition for public libraries. Francis et al. [25] develop models to assist a large suburban library system to manage its vehicle fleet and optimize operations under budget constraints. The objective function balances travel time incurred by the delivery agency and the service benefit to the libraries served. A similar, smaller interlibrary loan system in Ohio, studied in Min [68], differs from the current example in that all libraries are visited daily.

**2.2.4. Public Education.** Thanassoulis and Dunstan [85] show how data envelopment analysis (DEA) can be used to guide secondary schools to improved performance through role-model identification and target setting in a way that recognizes the multioutcome nature of the education process and reflects the relative desirability of improving individual outcomes. The approach presented in the paper draws from a DEA-based assessment of the schools of a local education authority carried out by the authors. Mar Molinero [64] develops recommendations for school closures in a region with declining school-age population using techniques to measure the similarity of school catchment areas that are input to a multidimensional scaling analysis that identifies socio-economic characteristics of these areas. Recommendations to close high-cost, low-performing schools that serve most-disadvantaged regions are contrasted with political opposition from local communities that seek to preserve local educational opportunities. Bowerman [14] formulates a multiobjective model for urban school-bus routing that addresses efficiency and equity jointly, and develops a two-phase approach combining student clustering and route generation. Taylor et al. [84] describe forecasting models for school attendance and optimization models for public-school locations and attendance boundaries, which reflect detailed knowledge of school administrators, elected representatives, and planners. These models address the need for racial balance across schools to minimize the need for busing, and have increased the confidence of community stakeholders in the school planning process, as measured by decreased political opposition to siting plans and increased passage rate of funding referenda.

## 2.3. Community Development

**2.3.1. Housing.** Kaplan [49], Kaplan and Amir [53], and Kaplan and Berman [54] formulate and solve math programs related to production-scheduling problems to design policies for relocating families in public-housing communities undergoing renovations to minimize total development time while ensuring that as few families as possible are displaced from public housing into private markets. Kaplan [50] uses queuing theory to evaluate the impacts of race-based versus nonrace-based tenant-assignment policies in public housing on levels of racial segregation and waiting times for available units. Forgionne [24] describes a decision-support system for assessing the army's needs for on-base new construction or off-base leased housing, determined in part by estimates of the level of off-base affordable housing available to its personnel.

Johnson and Hurter [45] generate alternative potential allocations of households using rental vouchers to Census tracts across a county to jointly optimize measures of net social benefit and equity, subject to constraints on programmatic and political feasibility. Estimates of dollar-valued impacts of subsidized housing are derived from models adapted from housing economics that use observations of actual households. Johnson [43] solves a multiobjective model for location of project-based affordable renter- and owner-occupied housing to optimize social efficiency and equity measures. Objective functions and structural parameter values are derived from discussions with nonprofit housing providers. Johnson [44] estimates structural parameters for math programming-based models for affordable housing design using microeconomic models of the firm and of the consumer, and statistical methods such as forecasting and factor analysis. Observations of housing units, households, and housing projects are provided by community-based nonprofit housing developers.

Johnson [39] presents a Web-accessible prototype of a SDSS for individual housing mobility counseling using multiple decision models and reflecting the needs of housing clients, counselors, and landlords. Johnson [42] provides a research framework for a professional-quality housing counseling SDSS and provides evidence derived from field research that typical subsidized housing program participants can make productive use of quantitative decision models. Both of these papers are discussed in greater detail in §3.

**2.3.2. Community/Urban Planning.** Norman and Norman [72] evaluate case studies of public art installations in Japan and the UK to support the concept of centralized, model-based decision making in the provision of public amenities. Patz et al. [74] develop decision models to design land-use strategies in a European context, allowing especially for existing historic buildings and districts. Jung et al. [48] develop a planning model for post-disaster reconstruction planning in urban areas inspired by Hurricane Katrina, which struck the U.S. Gulf Coast in 2005. In this model, small neighborhoods are aggregated into districts that share the same land-use designation (e.g., human use or passive use) in order to jointly optimize measures of environmental protection, social impacts, and spatial integrity.

**2.3.3. Transportation.** Vlahos et al. [93] develop a framework for computer-aided transportation planning that accommodates multiple, possibly antagonistic, stakeholders for the purposes of presentation, analysis, and judgment. A case study of this methodology emphasizes how community-based participants can collaborate with government and elected officials in developing a controversial transportation strategy. Murray and Davis [69] use community-level data on socio-economic status, public transport access, and public transport need to determine differential impacts on communities of alternative public transit investments.

## 2.4. Public Health and Safety

**2.4.1. Public Health.** Kaplan [51] estimates changes in HIV-infection rates associated with an experimental needle-exchange program using maximum likelihood change-point models and continuous-time Markov processes. Both models are estimated using observations of the HIV status of used needles provided by injection drug users. Aaby et al. [1] present models to improve pre-contagion planning for clinics that dispense medications and vaccines using discrete-event simulations, queuing models, and capacity planning models. Griffin et al. [32, 33] have developed a decision model based on the maximum covering facility location problem to identify centroids of U.S. counties at which community health centers (CHC) may be sited in order to maximize the weighted demand for CHC services, subject to constraints on funds available for fixed and variable costs and facility capacities. Their model qualifies as “community-based” in two ways: first, the care with which they have analyzed available health-care data to impute localized measures of health services demand, especially for medically underserved populations, and second, evidence provided that recommendations are fairly insensitive to smaller spatial units of analysis such as Census tracts. They find significant improvements in a variety of performance measures, including total encounters, cost per encounter, and number of uninsured persons served.

Models to address food insecurity include Chou and Zheng [20] and Lien et al. [61]. Chou and Zheng study a bread delivery problem where unsold bread from bakeries is delivered by volunteers to needy families. In their problem, demands are fixed and supply is random. They show how incorporating flexibility, meaning allowing volunteers to deliver to multiple families, can reduce the amount of wasted bread. Lien, Iravani, and Smilowitz develop mathematical models and solution methods for perishable food distribution from donors to agencies that address vehicle-routing problems (assigning donors and agencies to routes and sequencing stops within each route) and inventory-allocation problems (determining the amount to distribute to each agency). In contrast to cost-minimizing objectives, which can lead to inequitable solutions, a novel service-based maximin fill-rate objective incorporates fairness and agency sustainability, albeit at the cost of more complex modeling and solution techniques. This latter paper is discussed in greater detail in §3.

**2.4.2. Criminal Justice.** Brown and Liu [16] develop crime forecasts to support tactical operations, in particular the presence and intensity of local “hot spots” using a multivariate prediction model that relates the features in an area to the predicted occurrence of



crimes through the preference structure of criminals. Xu and Chen [98] use link analysis techniques based on shortest-path algorithms to identify networks of offenders in areas such as narcotics violation, terrorism, and kidnapping and find differing levels of effectiveness depending on the shortest-path search strategy. Johnson [40] develops decision models to select sites for community correction centers using multiobjective math programming and multicriteria decision analysis. These models reflect field research with stakeholders and propose siting strategies that account for neighborhood-level amenities that may be associated with successful re-entry to civilian life as well as concerns regarding potential re-offending. Blumstein [11] offers a comprehensive review of OR/MS applications to law enforcement and criminology, focused on policy modeling but reflecting close examination of community-level dynamics that motivate more stylized analytical approaches.

**2.4.3. Emergency Services.** The literature on fire and emergency medical services provision is extensive and long-lived; Swersey [82] provides a comprehensive review. Models for location of emergency services facilities (e.g., Walker [94], Marianov and ReVelle [65], Hogan and ReVelle [36]) and end-user applications (RAND Fire Project [77]) are innovative in applying set covering models, maximum covering models, and backup coverage models, they typically do not address community-level or equity issues explicitly. However, Schilling et al. [81] formulated a multiple-objective maximum-covering-based model for fire-station location that balances population coverage, property value coverage, and area coverage, i.e., the equity/efficiency/effectiveness objectives defined as key to community-based OR applications. The police-sector-design problem, in which a study area is partitioned into regions with equal service characteristics, has been dominated by queuing models such as the Larson's [58], [59] hypercube model. Bodily [13] applied multiattribute utility theory to the hypercube model to incorporate preferences of citizens interested in service equity, police officers interested in workload equalization, and administrators interested in system efficiency.

**2.4.4. Hazardous/Undesirable Facilities.** The process by which undesirable, obnoxious or hazardous facilities are located, and prescriptive models for doing so, contain elements of commonality with community-based OR. Kleindorfer and Kunreuther [57] provide a comprehensive overview of this domain and describe a detailed process for facility siting that incorporates community concerns. Gregory et al. [31] present policies by which community stakeholders can be fairly compensated for accepting location of undesirable facilities.

There are a variety of mathematical programming-based models for hazardous facility siting in which local community concerns and equity objectives are central. Ratick and White [78] formulate a model that balances measures of facility-location costs, political opposition as a function of the population within a risk radius of a facility, and equity, an increasing function of the number of communities perceived by any particular community as bearing equivalent risks associated with proximity to a facility. Erkut and Newman [22] extend the scale opposition factor within the model of Ratick and White by representing equity as a continuous (rather than discrete) function that increases in the distance between population centers, as well as decreasing in facility size. Murray et al. [70] choose locations for undesirable facilities to maximize the total population that is outside the "impact radius" of sited facilities, or, in a variant, maximize the total weighted population that is within the impact radius of at most one facility.

Merkhofer and Keeney [67] use decision theory to choose sites for a nuclear-waste repository while addressing community concerns, whereas Erkut and Moran [21] apply Analytic Hierarchy Process for location of municipal landfills, explicitly addressing a wide range of community-level factors that influence final rankings.

## 2.5. Nonprofit Management

Baker Werth [6] describes the development of a decision-support division within a county human-services agency to support strategic management and accountability services for local government. Vericourt and Lobo [92] model of a nonprofit's decision to partake in for-profit activities. For-profit activities often generate future revenue for nonprofit work at the expense of current resources. They show that under certain conditions the optimal investment decision in for-profit activities is of threshold type.

## 2.6. Lessons Learned

The past 15 years has seen a significant growth in research that is consistent with our definition of community-based OR. Increasingly, OR/MS researchers are aware of the need to incorporate socio-economic and political concerns directly into their planning models, rather than assert, as Gregg et al. [30] do in a stochastic programming model of facility location applied to library closings, that "political factors can be incorporated by means of user intervention" (p. 90). The models discussed in this section also reflect a desire to apply detailed understanding of stakeholder needs to quantitative models, which can yield actionable, policy-relevant prescriptions. These efforts stand in contrast to some research outputs associated with UK-style community OR and early U.S. efforts in this domain that focused more on community engagement and community efficacy. Finally, many of the models discussed in this section cross disciplinary boundaries, for example adapting models from other domains for purposes of decision making, or using research evidence outside of OR/MS to justify model-building efforts. However, more needs to be done to ensure that novel and innovative community-focused planning models are implemented in the field and their impacts on stakeholder groups evaluated rigorously. In this sense, the record of real-world impacts represented by models of the type presented in Larson and Odoni's *Urban Operations Research* [60] serve as a benchmark for community-based OR.

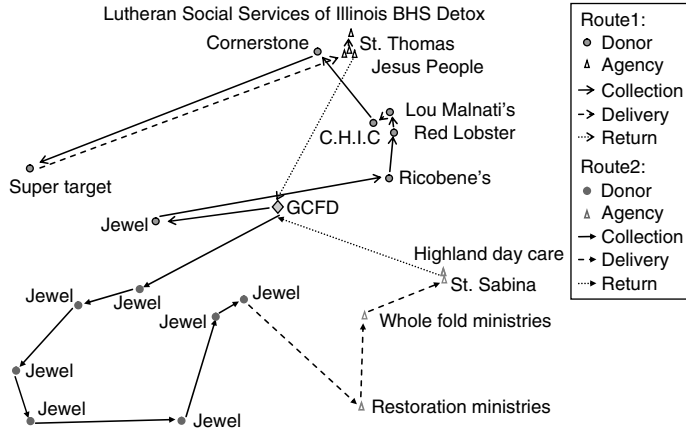
## 3. Applications

### 3.1. Food Security

In 2005, thirty-six million Americans suffered from hunger (U.S. Department of Agriculture [87]). Twenty-five million of these Americans rely on America's Second Harvest (ASH) and their network of pantries, shelters, and soup kitchens for food (ASH [4]). The largest suppliers to these agencies are regional and local food banks. Food banks are large-scale distribution centers that collect, store, and distribute food. Much of this food is donated by various sources of surplus food (e.g., supermarkets and grocery chains). According to the U.S. Department of Agriculture [88], about 96 billion pounds of food are wasted each year in the United States. The goal of ASH and the agencies in their network is to match surplus food with those in need. This matching is a large-scale distribution and inventory-management problem that occurs each day at thousands of nonprofit agencies across the country. Much research has been conducted on related supply-chain problems in commercial settings where the goal of such systems is either to maximize profit or minimize cost. Little work, however, has been conducted in nonprofit applications. In such settings, the objectives are often more difficult to quantify because issues such as equity and sustainability must be considered, yet efficient operations are still crucial.

The Greater Chicago Food Depository (GCFD) is an active ASH member. According to a recent study by the GCFD and ASH [29], 500,000 people in the Chicago region are served by the GCFD each year. One program run by the GCFD is the Food Rescue Program (FRP), which distributes perishable food from donors (e.g., supermarkets and restaurants) to agencies (e.g., shelters and soup kitchens). Over 80 donors and 100 agencies participate in the FRP, which moves over four million pounds of food annually.

FIGURE 1. Food rescue program sample routes.



The FRP operates five truck routes, each visiting between three and 17 donors and between two and 11 agencies daily. Two sample routes are shown in Figure 1. Routes begin at the depot, collect food from the donors, and distribute donations to the agencies. The frequency of visits to a location over the course of a month depends on the supply of the donor and the need of the agency. Routes are scheduled weeks in advance and remain fairly regular to facilitate driver familiarity with the routes they perform. Donation amounts and food demand are unknown until observed on the driver's arrival. The donation amounts depend on daily sales at the donor location; food demands depend on available storage and budget at the agencies. Drivers collect the full donation available at a donor. The allocation of food to agencies is left to the discretion of the driver who tries to satisfy an agency's demand while reserving supply for the remaining agencies on the route. Agencies are charged 4 cents per pound of food and view the FRP as a supplement to their main food acquisition operations.

Ongoing work (Lien et al. [61]) has focused on the design of inventory and routing policies for the FRP. The aim of this project is to develop mathematical models and solution methods for related vehicle-routing problems (assigning donors and agencies to routes and sequencing stops within each route) and inventory-allocation problems (determining the amount to distribute to each agency). In the private sector, related sequential inventory-allocation decisions are often made to either maximize revenue or minimize costs. Although cost-efficient operations remain desirable in the nonprofit sector, focusing purely on cost can lead to inequitable solutions. The GCFD seeks to provide all agencies with adequate resources in an equitable manner. The paper proposes an alternative service-based objective function (maximizing the minimum expected fill rate) that incorporates fairness and agency sustainability.

The use of an alternative objective function has important implications on the solution methods used to determine routing and inventory-allocation policies. The authors find that the useful mathematical properties of commercial distribution problems with profit-maximizing or cost-minimizing objectives (e.g., as in the multiperiod news-vendor problem) do not hold with the service-based objectives. As a result, new solution methods must be developed. The paper also proposes simple routing and inventory-allocation policies that perform well relative to more computer-intensive methods.

### 3.2. Affordable/Subsidized Housing

As of 2000, about 10% of the poor population in the U.S. lives in high-poverty neighborhoods (Census tracts with poverty rates of 40% or greater). An overall decrease in concentrated

poverty between 1990 and 2000 has been accompanied by stagnant or increasing levels of poverty concentration in the Northeastern and Western regions of the U.S., and increasing numbers of poor families living in communities far away from the centers of America's cities (Joint Center for Housing Studies of Harvard University [47]). Housing mobility, i.e., relocation of low-income families from distressed communities to more affluent, opportunity-rich communities, has been one goal of affordable and subsidized housing policy.

The Moving to Opportunity Program for Fair Housing (MTO), a national demonstration intended to evaluate outcomes of housing mobility-program participants, offers the possibility of significant improvements in living conditions and life outcomes for very low-income families who relocate using subsidies from the Housing Choice Voucher Program (HCVP). Over 6,000 families have participated in the experiment to date. Such areas for improvements in life outcomes include: risk of criminal victimization, housing quality, and mental health (U.S. Department of Housing and Urban Development [89]). MTO and HCVP is the inspiration for development of a prototype spatial decision-support system (SDSS) for housing-mobility counseling that might enable programs like MTO to be implemented on a large scale. This SDSS, called the Pittsburgh Housing eCounselor (<http://www.housingecounselor.org>) (Johnson [39]) is intended to assist housing counselors, clients, and landlords in connecting low-income families to good-quality housing in opportunity-rich neighborhoods.

This research confronts a number of gaps in current knowledge. First, there is little research on the actual decision-making process by which housing mobility program participants (or HCVP clients) search for housing. Second, the extent to which public housing authorities and other housing service agencies can assist creative decision making by clients who may face multiple life challenges is limited by a lack of technical knowledge on information technology (IT)-assisted decision making and the necessary IT infrastructure. Last, little is known about the ability of low-income families to frame and solve difficult problems using decision models and IT.

The Pittsburgh Housing eCounselor guides users through the process of identifying candidate housing units and neighborhoods, and ranking these candidate sites with two alternative multicriteria decision models (MCDM). The eCounselor uses Keeney's [56] value-focused thinking method to enable clients and counselors together to identify characteristics of housing units and neighborhoods that are important to the client and generate a subset of housing units and/or neighborhoods based on user-defined criteria. This process is represented by Figure 2, in which users start the search by identifying candidate neighborhoods or, alternatively, candidate housing units, through appropriate queries and rank candidate destinations using MCDM. Given these candidates for relocation, users choose acceptable housing units or, alternatively, neighborhoods, thus creating a set of acceptable housing units in acceptable neighborhoods suitable for site visits.

Neighborhood characteristics are represented with spatial data describing demographic, employment, and housing characteristics of Allegheny County, PA and displayed in a Web browser. Figure 3 shows an example of this spatial data display, for fair housing complaints in Allegheny County.

Housing unit characteristics are represented with tabular data describing actual housing units available for rent in the city of Pittsburgh and displayed in a Web browser similarly.

Neighborhoods or housing units can be ranked using one of two MCDMs: elimination by aspects (i.e., simple sort), in which users rank candidates in ascending order according to attribute values, or PROMETHEE (Brans and Vincke [15]), in which users specify the form of preference functions that measure the extent of a users' preference for one alternative over another with respect to the difference in performance of any pair of alternatives according to a single attribute. For example, a user could specify an increasing preference for one alternative over another as a linear function of the difference in crime rates of two neighborhoods.

FIGURE 2. Counseling support system destination search algorithm.

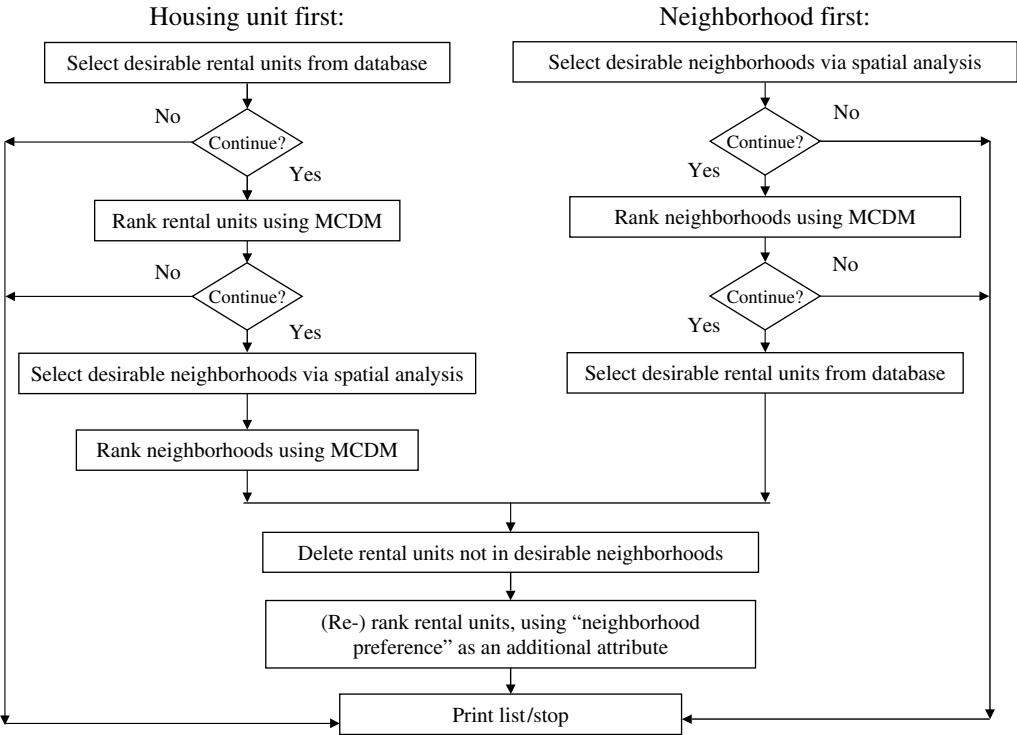
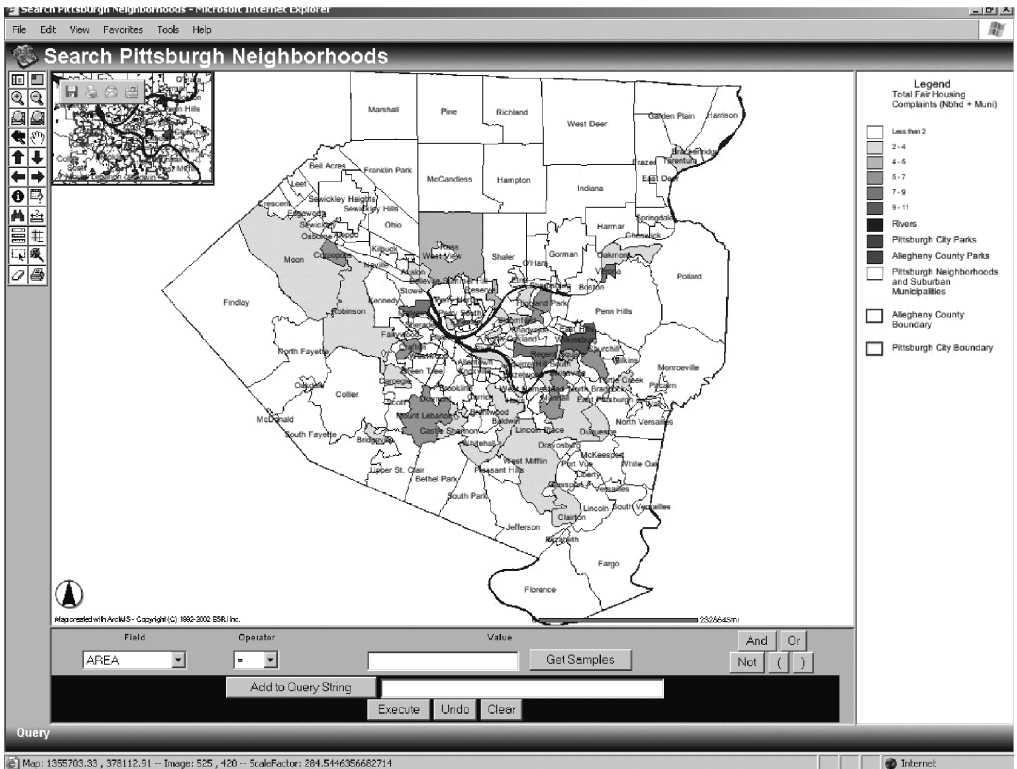


FIGURE 3. Search Pittsburgh neighborhoods—fair housing complaints.



The SDSS project has motivated subsequent field research to better understand the ability of typical clients to make use of various MCDMs (Johnson [42]). A sample of eight housing-voucher clients of a local public-housing authority used a custom application to rank hypothetical neighborhoods using three MCDMs: simple additive weighting using rank sum weights (Malczewski [62], p. 199), Analytic Hierarchy Process (AHP; Saaty [80]), and PROMETHEE. We found that representative assisted-housing clients appear to appreciate the increased insight provided by more analytically demanding MCDMs such as PROMETHEE, and that neighborhood rankings were largely dissimilar across the three MCDMs. Thus, this research is supportive of the notion of model-based decision methods as a tool for improving housing counseling, and provides suggestive evidence that a range of MCDMs may allow clients to choose the application that is best suited to their skills and preferences.

### 3.3. Discussion

These case studies have demonstrated the potential of diverse methods in OR/MS to address important problems of a local nature whose solutions may provide direct benefits to traditionally underserved or underrepresented populations. Equity is addressed in the food distribution application through an objective function that accommodates the needs of as many food pantries as possible; the housing relocation SDSS incorporates equity by allowing clients themselves to define criteria and alternatives consistent with their own preferences, while enlarging their choice sets to the greatest extent possible. Although the two applications are inspired by region-specific policy problems, the solution methods may be applied to many other regions. Finally, both applications reflect a concern with the ability of local organizations to implement the models' recommendations: the food distribution model uses heuristic solution methods and the housing relocation SDSS has a user interface that guides clients' interactions with decision models.

## 4. Conclusion

In this chapter we have described a subfield of public-sector OR called community-based operations research that addresses topics of interest to researchers and practitioners for over 30 years and is known to different audiences in the U.S. and UK under a variety of names. Common in our definition of community-based OR is a localized focus, a high degree of ambiguity, complexity, and controversy, and an emphasis on stakeholder groups that are traditionally underrepresented or underserved by social and political systems of interest to the modeler. Also, we emphasize the need for modelers to combine a focus on inclusion of individual actors and community-based organizations in defining and solving the problem (the traditional focus of UK-style "community OR") with an emphasis on quantitative models that generate actionable and policy-relevant recommendations. We have provided case studies in food security and affordable/subsidized housing that are consistent with the various components of the definition above.

Based on our review of the research literature and resources within higher education, we believe there is an opportunity to increase the influence of community-based OR on education, scholarship, and practice. Most importantly, we advocate an increase in the number of courses in undergraduate and graduate programs in business administration, industrial engineering/operations research, and public affairs that address special characteristics of nonprofit and government organizations and the decision problems they face. These courses should not be limited to project or capstone courses that have nonprofit organizations as clients, but theory-based lecture courses as well. These courses should provide: greater emphasis on cross-disciplinary evidence to support OR/MS modeling, multidisciplinary solution approaches that emphasize less complex and/or qualitative solution methods, field research to generate realistic problem instances, and low-cost IT platforms for model and

service delivery. Such courses could leverage the considerable teaching resources associated with spreadsheet-based management science rather than specialized mathematical modeling applications.

Also, we recommend that professional societies increase the returns to community-based OR research and practice. This could be done by encouraging workshops and conference sessions devoted to community-based OR, greater awareness within the profession of public policy/equity implications of models and applications, more support for cross-disciplinary research that provides an evidentiary basis for community-based OR models, and increased emphasis on international public-sector applications of community-based OR.

There are many potential application areas for community-based OR. These include: location of businesses and services in low-income, especially urban and predominately minority communities, such as financial services for the unbanked; decision-support systems for system-level redesign for mass transit, public schools, and senior services; and community revitalization through reuse of vacant lots and abandoned buildings. Finally, given the ubiquity of the Internet, and recent decreases in the “digital divide” across race, ethnic, and income categories, there are opportunities for community-focused individual decision applications that use the Internet as a facilitator. Examples of these applications include identifying health and financial impacts of alternative food purchase strategies, especially for low-income families, behavior change for energy reduction, and matching needs and locations of low- and moderate-income families seeking family-support services with service providers.

## Acknowledgments

The authors thank Louis Luangkesorn at RAND Corporation, for initial inspiration to explore this domain in a formal way, and Janet Hunziker at the National Academy of Engineering for providing opportunities to explore this area more deeply. They also thank Philip Akol, Robert Lien, and Patrick Mallory for their research assistance and Mark Daskin, Ed Kaplan, and Julie Swann for valuable literature references. This tutorial benefited greatly from the comments of Al Blumstein, two anonymous reviewers, and the editor of this volume. This work was supported in part by the National Science Foundation Faculty Early Career Development (CAREER) Program, (CAREER: Public-Sector Decision Modeling for Facility Location and Service Delivery (Michael Johnson) and CAREER: Strategies to Improve Goods Movement: Operational Choice in Routing (Karen Smilowitz)).

## References

- [1] K. Aaby, J. W. Herrmann, C. S. Jordan, M. Treadwell, and K. Wood. Montgomery county’s (Maryland) public health service uses operations research to plan emergency mass dispensing and vaccination clinics. *Interfaces* 36(6):569–579, 2006.
- [2] R. L. Ackoff. A black ghetto’s research on a university. *Operations Research* 18:761–771, 1970.
- [3] R. L. Ackoff. The future of operational research is past. *Journal of the Operational Research Society* 30:93–104, 1979.
- [4] America’s Second Harvest. Face of hunger in your community. [http://www.secondharvest.org/who\\_we\\_help/hunger\\_facts.html](http://www.secondharvest.org/who_we_help/hunger_facts.html), 2005.
- [5] S. M. Bajgier, H. D. Maragah, M. S. Saccucci, A. Verzilli, and V. R. Prybutok. Introducing students to community operations research by using a city neighborhood as a living laboratory. *Operations Research* 39(5):701–709, 1991.
- [6] J. Baker Werth. Getting to the bottom line in local government: How San Diego county’s health and human services agency uses decision support techniques to help agency executives make better decisions. *The Public Manager* 32(2):21–24, 2003.
- [7] B. Balcik and B. Beamon. Distribution network design for humanitarian relief chains. Working paper, University of Washington, Seattle, WA, 2005.
- [8] E. Bardach. *A Practical Guide for Policy Analysis: The Eightfold Path to More Effective Problem Solving*. Chatham House Publishers, Seven Bridges Press, New York, 2000.

- [9] J. J. Bartholdi, R. L. Collins, L. Platzman, and W. H. Warden. A minimal technology routing system for meals on wheels. *Interfaces* 13(3):1–8, 1983.
- [10] B. Beamon and S. Kotleba. Inventory modelling for complex emergencies in humanitarian relief operations. *International Journal of Logistics* 9(1):1–18, 2006.
- [11] A. Blumstein. An OR missionary's visits to the criminal justice system. *Operations Research* 55(2):14–23, 2007.
- [12] A. Blumstein, F. P. Rivara, and R. Rosenfeld. The rise and decline of homicide—and why. *Annual Review of Public Health* 21:505–541, 2000.
- [13] S. Bodily. Police sector design incorporating preferences of interest groups for equality and efficiency. *Management Science* 24(12):1301–1313, 1978.
- [14] R. Bowerman. A multi-objective optimization approach to urban school bus routing: Formulation and solution method. *Transportation Research A* 29A(2):107–123, 1995.
- [15] J. P. Brans and P. Vincke. A preference ranking organisation method (the PROMETHEE method for multiple criteria decision making). *Management Science* 31(6):647–656, 1985.
- [16] D. E. Brown and H. Liu. Criminal incident prediction using a point-pattern-based density model. *International Journal of Forecasting* 19(4):603–622, 2003.
- [17] Bureau of Justice Statistics. Prison and jail inmates at midyear 2005. U.S. Department of Justice, Office of Justice Programs. <http://www.ojp.usdoj.gov/bjs/pub/pdf/pjim05.pdf>, 2006.
- [18] A. Campbell, D. Vandebussche, and W. Hermann. Routing in relief efforts. Working paper, University of Iowa, Ames, IA, 2006.
- [19] P. B. Checkland. The application of systems thinking in a real-world problem situation: The emergence of soft systems methodology. M. C. Jackson and P. Keys, eds. *New Directions in Management Science*. Gower, Aldershot, 87–96, 1987.
- [20] M. C. T. Chou and H. Zheng. Process flexibility revisited: Graph expander and food-from-the-heart. Working paper, Business School, National University of Singapore, Singapore, 2005.
- [21] E. Erkut and S. R. Moran. Locating obnoxious facilities in the public sector: An application of the analytic hierarchy process to municipal landfill siting decisions. *Socio-Economic Planning Sciences* 25(2):89–102, 1991.
- [22] E. Erkut and S. Neuman. A multiobjective model for locating undesirable facilities. *Annals of Operations Research* 40:209–227, 1992.
- [23] D. Erlenkotter. Facility location with price-sensitive demands: Private, public, and quasi-public. *Management Science* 24(4):378–386, 1977.
- [24] G. A. Forgy. Forecasting army housing supply with a DSS-delivered econometric model. *Omega* 24(5):561–576, 1996.
- [25] P. Francis, K. Smilowitz, and M. Tzur. The period vehicle routing problem with service choice. *Transportation Science* 40(4):439–454, 2006.
- [26] S. I. Gass. Public sector analysis and operations research/management science. S. M. Pollock, M. H. Rothkopf, and A. Barnett, eds. *Operations Research in the Public Sector*. North-Holland, Amsterdam, The Netherlands, 23–46, 1994.
- [27] B. Gates. Remarks. Washington learns educational summit. November 13, Seattle, WA, <http://www.gates-foundation.org/MediaCenter/Speeches/Co-ChairSpeeches/BillgSpeeches/BGSpeechWashingtonLearns-061113.htm>, 2006.
- [28] W. Gorr, M. Johnson, and S. Roehrig. Facility location model for home-delivered services: Application to the meals-on-wheels program. *Journal of Geographic Systems* 3:181–197, 2001.
- [29] Greater Chicago Food Depository and America's Second Harvest. The national hunger study: Chicagoprofile. Chicago, IL. [http://www.chicagosfoodbank.org/site/DocServer/HungerStudy\\_9.06.pdf?docID=301](http://www.chicagosfoodbank.org/site/DocServer/HungerStudy_9.06.pdf?docID=301), 2005.
- [30] S. R. Gregg, J. M. Mulvey, and J. Wolpert. A stochastic planning system for siting and closing public service facilities. *Environment and Planning A* 20:83–98, 1988.
- [31] R. Gregory, H. Kunreuther, D. Easterling, and K. Richards. Incentives policies to site hazardous waste facilities. *Risk Analysis* 11(4):667–675, 1991.
- [32] P. M. Griffin, C. R. Sherrer, and J. L. Swann. Optimization of community health center locations and service offerings with statistical need estimation. Working paper, Georgia Institute of Technology, School of Industrial and Systems Engineering, Atlanta, GA, 2006.
- [33] P. M. Griffin, C. R. Sherrer, and J. L. Swann. Access through community health centers or coverage through medicaid: A geographical and mathematical analysis of the State of Georgia. Working paper, Georgia Institute of Technology, School of Industrial and Systems Engineering, Atlanta, GA, 2007.



- [34] A. Haghani and S.-C. Oh. Formulation and solution of a multi-commodity multi-modal network flow model for disaster relief operations. *Transportation Research A* 30(3):231–250, 1996.
- [35] C. J. Heinrich. Measuring public sector performance and effectiveness. B. G. Peters and J. Pierre, eds. *Handbook of Public Administration*. SAGE, London, UK, 25–37, 2003.
- [36] K. Hogan and C. ReVelle. Concepts and applications of backup coverage. *Management Science* 32(11):1434–1444, 1986.
- [37] M. C. Jackson. Some methodologies for community operational research. *Journal of the Operational Research Society* 39:715–724, 1988.
- [38] H. Jia, F. Ordonez, and M. Dessouky. A modeling framework for facility location of medical services for large-scale emergencies. Working paper, University of Southern California, USC-ISE working paper 2005-01, Los Angeles, CA, 2005.
- [39] M. P. Johnson. Spatial decision support for assisted housing mobility counseling. *Decision Support Systems* 41(1):296–312, 2005.
- [40] M. P. Johnson. Decision models for location of community corrections centers. *Environment and Planning B: Planning and Design* 33(3):393–412, 2006.
- [41] M. P. Johnson. OR/MS for public-sector decision making with limited resources: Values, evidence, and methods. Presented at INFORMS Fall National Conference, November 5, 2006, Pittsburgh, PA, 2006.
- [42] M. P. Johnson. Can a spatial decision support system improve low-income service delivery? Working paper, Carnegie Mellon University, Heinz School of Public Policy and Management, Pittsburgh, PA, 2006.
- [43] M. P. Johnson. Planning models for affordable housing development. *Environment and Planning B: Planning and Design* 34:501–523, 2007.
- [44] M. P. Johnson. Economic and statistical models for affordable housing policy design. Working paper, Carnegie Mellon University, Heinz School of Public Policy and Management, Pittsburgh, PA, 2007.
- [45] M. P. Johnson and A. P. Hurter. Decision support for a housing relocation program using a multi-objective optimization model. *Management Science* 46(12):1569–1584, 2000.
- [46] M. P. Johnson, W. L. Gorr, and S. Roehrig. Location of elderly service facilities. *Annals of Operations Research* 136(1):329–349, 2005.
- [47] Joint Center for Housing Studies of Harvard University. The state of the nation’s housing 2006. <http://www.jchs.harvard.edu/publications/markets/son2006/son2006.pdf>, 2006.
- [48] C. Jung, M. P. Johnson, and J. Williams. Mathematical models for reconstruction planning in urban areas. Working paper, Carnegie Mellon University, Heinz School of Public Policy and Management, Pittsburgh, PA, 2006.
- [49] E. H. Kaplan. Relocation models for public housing redevelopment programs. *Environment and Planning B* 13:5–19, 1986.
- [50] E. H. Kaplan. Analyzing tenant assignment policies. *Management Science* 33(3):395–408, 1987.
- [51] E. H. Kaplan. Probability models of needle exchange. *Operations Research* 43(4):558–569, 1995.
- [52] E. H. Kaplan. Adventures in policy modeling! Operations research in the community and beyond. *Omega* 36(1):1–9, 2008.
- [53] E. H. Kaplan and A. Amir. A fast feasibility test for relocation problems. *European Journal of Operational Research* 35(2):201–206, 1987.
- [54] E. H. Kaplan and O. Berman. OR hits the heights: Relocation planning at the orient heights housing project. *Interfaces* 18(6):14–22, 1988.
- [55] E. H. Kaplan and M. H. Merson. Allocating HIV prevention resources: Balancing efficiency and equity. *American Journal of Public Health* 92:1905–1907, 2002.
- [56] R. L. Keeney. *Value-Focused Thinking: A Path to Creative Decision making*. Harvard University Press, Cambridge, MA, 1992.
- [57] P. R. Kleindorfer and H. C. Kunreuther. Siting of hazardous facilities. S. M. Pollock, M. H. Rothkopf, and A. Barnett, eds. *Operations Research in the Public Sector*. North-Holland, Amsterdam, The Netherlands, 403–440, 1994.
- [58] R. Larson. A hypercube queueing model for facility location and redistricting in urban emergency services. *Journal of Computational Operations Research* 1(1):67–95, 1974.
- [59] R. Larson. Approximating the performance of urban emergency service systems. *Operations Research* 23(5):845–868, 1975.

- [60] R. C. Larson and A. R. Odoni. *Urban Operations Research*. Prentice-Hall, Englewood Cliffs, NJ, 1981.
- [61] R. Lien, S. Irvani, and K. Smilowitz. Sequential allocation problems for nonprofit agencies. Working paper, Northwestern University, Evanston, IL, 2007.
- [62] J. Malczewski. *GIS and Multicriteria Decision Analysis*. John Wiley & Sons, New York, 1999.
- [63] M. Mandell. Modelling effectiveness-equity trade-offs in public service delivery systems. *Management Science* 37(4):467–482, 1991.
- [64] C. Mar Molinero. Schools in Southampton: A quantitative approach to school location, closure and staffing. *Journal of the Operational Research Society* 39:339–350, 1988.
- [65] V. Marianov and C. ReVelle. The standard response fire protection siting problem. *INFOR* 29(2):116–129, 1991.
- [66] K. McCardle. O.R. for the public good. *OR/MS Today* 32(5):32–36, 2005.
- [67] M. W. Merkhofer and R. L. Keeney. A multiattribute utility analysis of alternative sites for the disposal of nuclear waste. *Risk Analysis* 7:173–194, 1987.
- [68] H. Min. The multiple vehicle-routing problem with simultaneous delivery and pick-up points. *Transportation Research A* 23(5):377–386, 1989.
- [69] A. T. Murray and R. Davis. 2001. Equity in regional service provision. *Journal of Regional Science* 41(4):557–600.
- [70] A. T. Murray, R. L. Church, R. A. Gerrard, and W.-S. Tsui. Impact models for siting undesirable facilities. *Papers in Regional Science* 77(1):19–36, 1998.
- [71] National Center for Educational Statistics. Percentile distribution of average reading and mathematics scores of 4th- and 8th-grade public school students and the percentage of students at each achievement level, by school location: 2003. <http://nces.ed.gov/programs/coe/2005/section2/table.asp?tableID=257>, 2006.
- [72] E. Norman and J. Norman. Operational research and the management of public art projects. *OR Insight* 14(1):14–23, 2001.
- [73] R. Parry and J. Mingers. Community operational research: Its context and future. *Omega* 19:577–586, 1991.
- [74] R. Patz, J. Spitzner, and C. Tammer. Decision support for location problems in town planning. *International Transactions in Operational Research* 9(3):261–278, 2002.
- [75] S. M. Pollock and M. D. Maltz. Operations research in the public sector: An introduction and brief history. S. M. Pollock, M. H. Rothkopf, and A. Barnett, eds. *Operations Research in the Public Sector*. North-Holland, Amsterdam, The Netherlands, 5–6, 1994.
- [76] S. M. Pollock, M. H. Rothkopf, and A. Barnett, eds. *Operations Research in the Public Sector*. North-Holland, Amsterdam, The Netherlands, 1994.
- [77] RAND Fire Project. W. Walker, J. Chaiken, and E. Ignall, eds. *Fire Department Deployment Analysis*. Elsevier/North-Holland, New York, 1979.
- [78] S. J. Ratick and A. L. White. A risk-sharing model for locating noxious facilities. *Environment and Planning B* 15:165–179, 1988.
- [79] C. P. Rydell, J. P. Caulkins, and S. S. Everingham. Enforcement or treatment? Modeling the relative efficacy of alternatives for controlling cocaine. *Operational Research* 44:687–695, 1996.
- [80] T. L. Saaty. How to make a decision: The analytic hierarchy process. *European Journal of Operational Research* 48(1):9–26, 1990.
- [81] D. Schilling, D. Elzinga, J. Cohon, R. Church, and C. ReVelle. The TEAM/FLEET models for simultaneous facility and equipment sizing. *Transportation Science* 13:163–175, 1979.
- [82] A. J. Swersey. The deployment of police, fire, and emergency medical units. S. M. Pollock, M. H. Rothkopf, and A. Barnett, eds. *Operations Research in the Public Sector*. North-Holland, Amsterdam, The Netherlands, 151–200, 1994.
- [83] A. Taket and L. White. Doing community operational research with multicultural groups. *Omega* 22(6):579–588, 1994.
- [84] R. G. Taylor, M. L. Vasu, and J. F. Causby. Integrated planning for school and community: The case of Johnston county, North Carolina. *Interfaces* 29(1):67–89, 1999.
- [85] E. Thanassoulis and P. Dunstan. Guiding schools to improved performance using data envelopment analysis. *Journal of the Operational Research Society* 45(11):1247–1262, 1994.
- [86] U.S. Census Bureau. Poverty: 2005 highlights. Housing and household economic statistics division, Washington, D.C. <http://www.census.gov/hhes/www/poverty/poverty05/pov05hi.html>, 2006.

- [87] U.S. Department of Agriculture. Estimating and addressing America's food losses. Economic research service. Prepared by Linda Scott Kantor, Kathryn Lipton, Alden Manchester, and Victor Oliveira, Washington, D.C. <http://www.ers.usda.gov/publications/FoodReview/Jan1997/jan97a.pdf>, 1997.
- [88] U.S. Department of Agriculture. Household food security in the United States, 2005. Economic research service, food assistance & nutrition research program. Prepared by Mark Nord, Margaret Andrews and Steven Carlson, Washington, D.C. <http://www.ers.usda.gov/Publications/ERR29/ERR29.pdf>, 2005.
- [89] U.S. Department of Housing and Urban Development. Moving to opportunity for fair housing demonstration program: Interim impacts evaluation. Office of Policy Development and Research, Washington, D.C., 2003.
- [90] *U.S. News and World Report*. America's best colleges 2007. <http://colleges.usnews.rankingsandreviews.com/usnews/edu/college/rankings/rankindex.brief.php>, 2007.
- [91] *U.S. News and World Report*. America's best graduate schools 2008. <http://grad-schools.usnews.rankingsandreviews.com/usnews/edu/grad/rankings/rankindex.brief.php>, 2007.
- [92] F. Vericourt and M. Lobo. Resource and revenue management in nonprofit operations. Working paper, Fuqua School of Business, Duke University, Durham, NC, 2005.
- [93] N. J. Vlahos, A. Khattak, M. L. Manheim, and A. Kanafani. The role of teamwork in a planning methodology for intelligent urban transportation systems. *Transportation Research C* 2C:217–229, 1994.
- [94] W. Walker. Using the set covering problem to assign fire companies to fire houses. *Operations Research* 20(3):275–277, 1974.
- [95] B. Western. *Punishment and Inequality in America*. Russell Sage Foundation Publications, New York, 2006.
- [96] D. W. S. Wong and J. W. Meyer. A spatial decision support system approach to evaluate the efficiency of a meals-on-wheels program. *Professional Geographer* 45(3):332–341, 1993.
- [97] R. E. Woolsey. On doing well by doing good and an offer of free education. *Interfaces* 28(2):99–103, 1998.
- [98] J. J. Xu and H. Chen. Fighting organized crimes: Using shortest-path algorithms to identify associations in criminal networks. *Decision Support Systems* 38(3):319–493, 2004.

# Generating Robust Project Baseline Schedules

**Willy Herroelen**

Research Center for Operations Management, Katholieke Universiteit Leuven, B-3000 Leuven, Belgium, willy.herroelen@econ.kuleuven.be

**Abstract** Most research efforts in resource-constrained project scheduling assume a static and deterministic environment within which the precomputed baseline schedule will be executed. Project activities, however, may be subject to considerable uncertainty, which may lead to numerous schedule disruptions during project execution. In this tutorial, we discuss proactive project scheduling procedures for generating robust baseline schedules that are sufficiently protected against anticipated time and/or resource disruptions in combination with reactive policies that may be deployed to repair the baseline schedule during project execution.

**Keywords** project scheduling; uncertainty; stability; buffers

---

## 1. Introduction

In this tutorial we focus on proactive procedures for generating project baseline schedules that are sufficiently protected against disruptions that may be caused by uncertainties in the activity durations and/or resource availabilities. The proactive procedures can be used in combination with reactive procedures to be deployed when the baseline schedule, despite its protection, breaks.

### 1.1. The Project Baseline Scheduling Problem

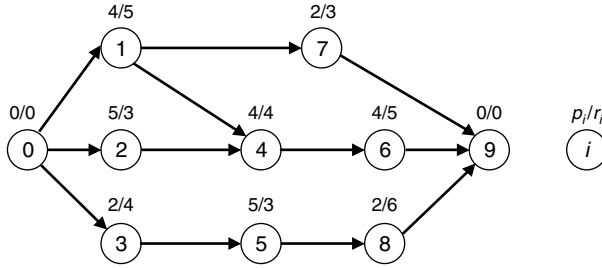
The vast majority of the project scheduling research efforts over the past several years has concentrated on the development of workable baseline schedules, assuming complete information and a static and deterministic environment. Such a *baseline schedule* (*preschedule*, *predictive schedule*) is traditionally constructed by solving the so-called *resource-constrained project scheduling problem* (RCPSP).

The RCPSP (problem  $m, 1|cpm|C_{\max}$  in the notation of Herroelen et al. [15]) involves the determination of activity starting times that satisfy both the zero-lag finish-start precedence constraints and the renewable resource constraints under the objective of minimizing the project duration (for reviews, we refer to Brucker et al. [7], Demeulemeester and Herroelen [9], Herroelen et al. [14], Kolisch and Padman [20]).

The deterministic RCPSP can be stated as follows. Consider a project network  $G(N, A)$  represented in activity-on-the-node representation format with dummy start node 0 and dummy end node  $n$ . All nondummy project activities have durations  $p_i$  and are subject to zero-lag finish-start precedence constraints on the elements of  $A$ : We require  $s_j \geq s_i + p_i$  if  $(i, j) \in A$ , with  $s_i$  the planned start time of activity  $i$ . Nondummy activities require an integer per period amount  $r_{ik}$  of one or more renewable resource types  $k$ ,  $k = 1, \dots, q$ , during their execution. All resource types  $k$  have a per-period capacity  $a_k$ . The objective is to derive a precedence and resource feasible baseline schedule  $\mathcal{S}^B = (s_0, s_1, \dots, s_n)$  of activity start times that minimizes the duration of the project.

A project network example is shown in Figure 1 (Van de Vonder [37]). For each activity, the duration and per period requirement for a single renewable resource are shown above

FIGURE 1. Project network example.



the corresponding node. The single renewable resource type is assumed to have a per period availability  $a = 10$ .

Conceptually, the RCPSP can be formulated as follows:

$$\text{minimize } s_n \quad (1)$$

$$\text{subject to } s_i + p_i \leq s_j \quad \forall (i, j) \in A \quad (2)$$

$$s_0 = 0 \quad (3)$$

$$\sum_{i: i \in \mathcal{A}_t} r_{ik} \leq a_k \quad \text{for } k = 1, \dots, q \text{ and } t = 1, \dots, s_n. \quad (4)$$

The set  $\mathcal{A}_t$  that is used in Equation (4) denotes the set of activities that are in progress at time  $t$ . The objective function Equation (1) minimizes the start time of the dummy end activity and hence the duration of the project. Equation (2) expresses the finish-start zero-lag precedence relations, whereas Equation (3) forces the dummy start activity to start at time 0. Finally, Equation (4) expresses that at no time instant during the project horizon the resource availability may be violated. The formulation is conceptual: The linear program cannot be solved directly because there is no easy way to translate the set  $\mathcal{A}_t$  into a linear programming formulation. We refer to Demeulemeester and Herroelen [9] for a detailed discussion of mathematical programming formulations for the RCPSP. The deterministic RCPSP has been shown to be strongly  $\mathcal{NP}$ -hard (Blazewicz et al. [6]).

A minimum makespan schedule for the project network of Figure 1 is shown in Figure 2. The critical sequence, which determines the 15-period project duration is the chain  $\langle 0, 2, 4, 6, 8, 9 \rangle$ . Figure 3 shows the corresponding resource profile. A baseline schedule  $\mathcal{S}^B$ , such as the one given in Figure 2, serves a number of important functions (Aytug et al. [4], Mehta and Uzsoy [31], Wu et al. [47]). One of these is to provide internal visibility within the organization of the planned activity execution periods reflecting the requirements for the key staff, equipment, and other resources. The baseline schedule is also the starting point for communication and coordination with external entities in the company's inbound and outbound supply chain: It constitutes the basis for agreements with suppliers and subcontractors (e.g., for planning external activities such as material procurement and preventive

FIGURE 2. Minimum makespan schedule.

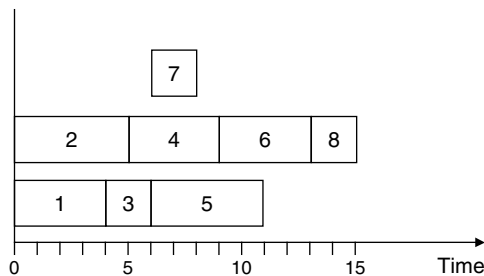
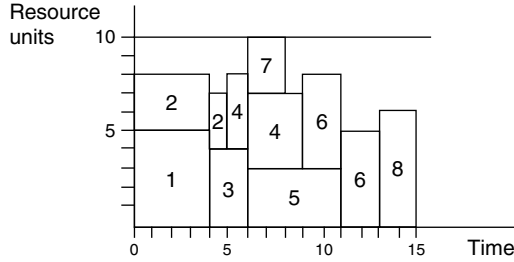


FIGURE 3. Resource profile for the minimum makespan schedule.



maintenance), as well as for commitments to customers (delivery dates). During execution, however, a project may be subject to considerable uncertainty, which may lead to numerous disruptions in the baseline schedule.

Many types of disruptions have been identified in the literature (we refer to Yu and Qi [48], Wang [46], and Zhu et al. [49]). Activities can take longer or shorter than expected, resource requirements or availability may vary, ready times and due dates may change, new activities may have to be inserted in the schedule, the project may have to be interrupted for a certain time, etc. In this tutorial we focus on schedule disruptions that may be caused by uncertainty in the duration of the project activities and/or in the availability of the renewable resources.

*Proactive/reactive project scheduling* procedures try to cope with schedule disruptions that may occur during project execution through the combination of a *proactive scheduling procedure* for generating a robust baseline schedule with a *reactive procedure* that is invoked when a schedule breakage occurs during project execution and the schedule needs to be repaired. Research in proactive/reactive project scheduling is growing steadily (for surveys in machine scheduling and project scheduling environments, see Aytug et al. [4], Herroelen and Leus [12, 13], and Vierra et al. [45]). The objective of this tutorial is to discuss promising proactive/reactive project scheduling procedures that may be deployed at the occurrence of disruptions that are caused by uncertain activity durations or uncertain resource availabilities.

The chapter is organized as follows. Section 2 distinguishes between the concepts of solution and quality robustness and defines a number of robustness measures. Section 3 focuses on exact and suboptimal proactive/reactive scheduling procedures under the assumption that the uncertainty stems from the activity durations. Both resource allocation and time-buffering proactive procedures are discussed that can be used in conjunction with exact and suboptimal reactive strategies that may be deployed during project execution upon schedule breakage. Section 4 concentrates on solution robust scheduling under resource availability uncertainty. We discuss an integrated proactive scheduling procedure to be used in combination with exact or suboptimal reactive scheduling methods. The last section presents some overall conclusions.

## 2. Proactive/Reactive Project Scheduling

Proactive/reactive scheduling involves a proactive and a reactive phase. During the proactive phase, a baseline schedule  $\mathcal{S}^B$  is constructed that accounts for statistical knowledge of uncertainty and anticipates disruptions. The underlying idea is to protect the schedule as much as possible against the distortions that may occur during the execution of the project. When disruptions occur during actual project execution, it may be necessary to call upon reactive scheduling procedures to modify the baseline schedule in response to these disruptions. The schedule actually executed after these modifications is called the *realized schedule*  $\mathcal{S}^R$  (Aytug et al. [4]). In general terms, a baseline schedule that is rather “insensitive” to

disruptions that may occur during project execution is called *robust*. The robustness concept has been used in many disciplines (see e.g., Kouvelis and Yu [21], Roy [34], and Billaut et al. [5]). Many different types of robustness have been identified in the literature.

In the next section we introduce proper definitions and measures.

## 2.1. Robustness Types and Measures

The robustness measure used can be single or composite. Two often used types of *single robustness measures* have been distinguished: quality robustness and solution robustness (Herroelen and Leus [13], Van de Vonder et al. [43]; for other typologies we refer to Sanlaville [35]).

**2.1.1. Solution Robustness or Schedule Stability.** Solution robustness or schedule stability refers to the *difference* between the baseline schedule  $\mathcal{S}^B$  and the realized schedule  $\mathcal{S}^R$ . This difference or *distance*  $\Delta(\mathcal{S}^B, \mathcal{S}^R)$  for a given execution scenario can be measured in a number of ways: the number of disrupted activities, the difference between the planned and realized activity start times, etc. Sanlaville [35] suggests to measure solution robustness as

$$\max_I \Delta(\mathcal{S}^B, \mathcal{S}^R), \quad (5)$$

the maximum difference between the baseline schedule  $\mathcal{S}^B$  and the realized schedule  $\mathcal{S}^R$  over the set of execution scenarios  $I$ . The objective of the proactive/reactive scheduling procedure then is to minimize the maximum distance between the baseline and the realized schedule.

Leus and Herroelen [29] suggest to measure the difference by the weighted sum of the absolute difference between the planned and realized activity start times, i.e.

$$\Delta(\mathcal{S}^B, \mathcal{S}^R) = \sum_{i \in N} w_i |S_i - s_i|, \quad (6)$$

where  $s_i$  denotes the planned starting time of activity  $i \in N$  in the baseline schedule  $\mathcal{S}^B$ ,  $S_i$  is a random variable denoting the actual starting time of activity  $i$  in the realized schedule  $\mathcal{S}^R$ , and the weights  $w_i$  represent the activity disruption cost per time unit, i.e., the nonnegative cost per unit time overrun or underrun on the start time of activity  $i$ . This disruption cost reflects either the difficulty in shifting the booked time window on the required resources (*internal stability*, or the difficulty in obtaining the required resources) or the importance of on-time performance of the activity (*external stability*). In practice, these penalties may be considerable. For example, the penalty of not meeting the delivery date of the renovated Berlaymont Building, housing the European Commission in Brussels (Belgium), was set to EUR 221,000 per month of delay (Kinnock [19]).

The objective of the proactive/reactive scheduling procedure is then to minimize  $\sum_{i \in N} w_i E|S_i - s_i|$  with  $E$  denoting the expectation operator; i.e., to minimize the weighted sum of the expected absolute difference between the planned and realized activity start times. It should be observed that the exact determination of the expected value of a function of the activity durations is unrealistic (Möhring [32], Leus and Herroelen [29]). Hagstrom [11] has shown for projects without resource constraints that even when every stochastic activity duration  $P_i$  can take only two discrete values, then computing the expected project duration and computing the probability that the project is finished by a given time instant, assuming an early-start schedule, is  $\#\mathcal{P}$  complete. For  $\mathcal{NP}$ -hardness proofs of several cases of the scheduling problem for stability for projects subject to a deadline and discrete disturbance scenario, we refer to Leus and Herroelen [30]. The objective function is usually determined using simulation (Igelmund and Radermacher [17], Leus and Herroelen [29], Stork [36]). The obtained objective function values will then be dependent on the simulated disruptions and

on the reactive procedure applied during the simulated project execution in order to repair the schedule upon breakage (Leon et al. [26]).

As will be illustrated later in this tutorial, solution robustness may also be evaluated using *surrogate* objective functions that are easier to compute (see also Aloulou and Portmann [1], Policella et al. [33], Deblaere et al. [8], Lambrechts et al. [22]–[25]).

**2.1.2. Quality Robustness.** Quality robustness refers to the insensitivity of the solution value of the baseline schedule to distortions. The ultimate objective of a proactive/reactive scheduling procedure is to construct a baseline schedule  $\mathcal{S}^B$  for which the solution value does not deteriorate when disruptions occur. The quality robustness is measured in terms of the value of the objective function  $z$ . In a project setting, commonly used objective functions are project duration (makespan), project earliness and tardiness, project cost, net present value, etc.

When stochastic data are available, quality robustness can be measured by considering the *expected value of the objective function*, such as the expected makespan  $E[C_{\max}]$ , the classical objective function used in stochastic resource-constrained project scheduling (Stork [36]).

It is logical to use the *service level* as a quality robustness measure, i.e., to maximize  $P(\mathbf{z} \leq z)$ , the probability that the solution value of the realized schedule stays within a certain threshold. As a result, we want to maximize the probability that the project completion time does not exceed the project due date  $d_n$ , i.e.,  $P(S_n \leq d_n)$ , where  $S_n$  denotes the actual starting time of the dummy end activity. Van de Vonder et al. [41] refer to this measure as the *timely project completion probability* (TPCP). It should be observed that even the analytic evaluation of this measure for a given schedule and in the presence of ample resource availability is very cumbersome, the PERT problem being  $\#\mathcal{P}$  complete (Hagstrom [11]).

Quality robustness can also be measured by comparing the solution value  $\mathbf{z}$  of the realized schedule obtained by the proactive/reactive scheduling procedure and the optimal solution value  $\mathbf{z}^*$  computed ex post by applying an exact procedure on the basis of the actually realized activity durations. Leus and Herroelen [28], for example, have used the percentage deviation of  $S_n$ , the project duration of the realized schedule, from the optimal makespan, computed by applying a branch-and-bound procedure on the basis of the actually realized activity durations as a measure of quality robustness.

**2.1.3. Composite Robustness Measures.** The robustness measures described above are all single measures. Also composite robustness objectives may be used (Hoogeveen [16]). Van de Vonder et al. [39] use the bicriteria objective  $F(P(S_n \leq d_n), \sum w_i E|S_i - s_i|)$  of maximizing the timely project completion probability and minimizing the weighted sum of the expected absolute deviation in activity starting times. The authors assume that the composite objective function  $F(.,.)$  is not known a priori, that the relative importance of the two criteria is not known from the outset, and that no clear linear combination is known that would reflect the preference of the decision maker. The analytic evaluation of the composite objective function is very cumbersome (as mentioned before, the PERT problem is  $\#\mathcal{P}$  complete (Hagstrom [11]) and the scheduling problem for stability is  $\mathcal{NP}$ -hard in the ordinary sense (Leus and Herroelen [30])). A natural way out is to evaluate the composite objective function through simulation.

### 3. Solution Robust Project Scheduling Under Activity Duration Uncertainty

#### 3.1. The Proactive/Reactive Scheduling Problem

We consider a project network  $G(N, A)$  represented in activity-on-the-node representation with dummy start node 0 and dummy end node  $n$ . All nondummy project activities are now assumed to have *stochastic activity durations*  $P_i$  and are subject to zero-lag finish-start precedence constraints on the elements of  $A$ : We require  $S_j \geq S_i + P_i$  if  $(i, j) \in A$ ,



with  $S_i$  the activity start time of activity  $i$  realized during project execution. Nondummy activities require an integer per period amount  $r_{ik}$  of one or more renewable resource types  $k$ ,  $k = 1, \dots, q$ , during their execution. All resource types  $k$  have a per-period capacity  $a_k$ . As before, the activity weight  $w_i$  denotes the marginal cost of a deviation between the realized start time  $S_i$  of activity  $i$  and its planned start time  $s_i$  in the baseline schedule. As mentioned earlier, these weights may include unforeseen storage costs, extra organizational costs, costs related to agreements with subcontractors, or just a cost that expresses the dissatisfaction of employees with schedule changes. We assume that  $w_0 = 0$ , whereas  $w_n$  denotes the cost of delaying the project completion beyond a predetermined deterministic due date  $d_n$ .

The proactive scheduling objective is to build a *solution robust* or *stable* precedence and resource feasible baseline schedule, the activity starting times of which are denoted by  $s_i$ . We assume in this chapter that stability is strived for by minimizing the stability function  $\sum_{i \in N} w_i E|S_i - s_i|$ , defined earlier in Equation (6).

### 3.2. Proactive Scheduling

We assume that a *two-phase procedure* is used for generating stable schedules. In the first phase, an input schedule is generated that is both precedence and resource feasible but is not intentionally protected against anticipated disruptions. Such a schedule can be generated by solving the underlying resource-constrained project scheduling problem (problem RCPSP or problem  $m, 1|cpm|C_{\max}$ ), using (deterministic) single-point estimates  $p_i$  for each  $P_i$ .

Two strategies may then be used in the second phase to increase the stability of the input schedule. One way is to aim at a robust resource allocation, i.e., to decide on a clever way in which the various resource units are transferred between the activities of the schedule. Another is to insert time buffers that should prevent as much as possible the propagation of distortions throughout the schedule.

Leus and Herroelen [30] have shown that under the assumption that a single activity may deviate from its preschedule duration (without knowing which) and a single disruption scenario per activity, the machine scheduling problem for stability is strongly  $\mathcal{NP}$ -hard for a single machine with unequal ready times or precedence constraints and for the case of a free number of parallel machines; the single-machine problem without ready times and precedence constraints is still ordinarily  $\mathcal{NP}$ -hard.

**3.2.1. Robust Resource Allocation.** Leus [27] and Leus and Herroelen [29] study the problem of generating a *robust resource allocation* for a *given* feasible baseline schedule  $S^B$ . They explore the fact that the search for an optimal resource allocation reduces to the search for a so-called *resource flow network* (which describes the routing of resources across the activities in the schedule) that exhibits desirable robustness characteristics. A branch-and-bound algorithm is developed that solves the robust resource allocation problem in exact and approximate formulations for the case of a *single renewable resource type* and exponential activity duration disruption lengths.

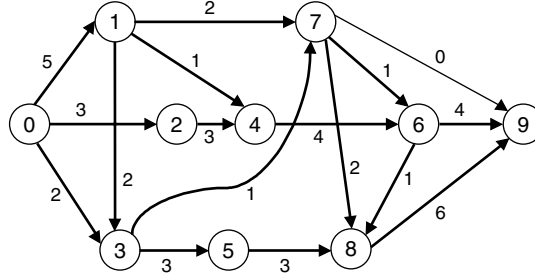
**Resource Flow Network.** Artigues and Roubellat [2] introduce the concept of a resource flow network in which the flow describes the resource units transferred among the activities. Let  $u_i = r_i$ ,  $\forall i \in N \setminus \{0, n\}$  and  $u_0 = u_n = a$ . A *resource flow*  $f$  associates with each activity pair  $(i, j) \in N \times N$  a value  $f_{ij} = f(i, j) \in \mathbb{N}$ . The flow values must respect the following constraints, which impose the conservation of flow in a node as well as a lower and upper bound on the flow

$$f(i, N) = u_i \quad \forall i \in N \setminus \{n\} \quad (7)$$

$$f(N, i) = u_i \quad \forall i \in N \setminus \{n\}, \quad (8)$$

$f_{ij}$  denotes the number of units of the single resource type transferred from activity  $i$  to activity  $j$ . For a flow  $f$ , let  $\Phi(f) = \{(i, j) \in N \times N \mid f_{ij} > 0\}$  denote the set of arcs carrying

FIGURE 4. Resource flow network for the example project.



nonzero flow. Let  $TA$  denote the transitive closure of  $A$ , meaning that  $(i, j) \in TA$  if a path exists from  $i$  to  $j$  in  $G(N, A)$ . The arcs  $\mathcal{X}(f) = \Phi(f) \setminus TA$  denote the flow carrying arcs that do not represent technological precedence constraints. In other words,  $\mathcal{X}(f)$  is a set of additional arcs inducing additional precedence constraints. For all  $X \subset N \times N$ ,  $G_X$  denotes the graph  $G(N, TA \cup X)$ . The flow  $f$  is a *feasible flow* if  $G_{\mathcal{X}(f)}$  is acyclic: The additional precedence constraints implied by  $\mathcal{X}(f)$  do not prohibit the realization of the project.

For the project network of Figure 1  $u_0 = u_9 = 10$ ,  $u_1 = 5$ ,  $u_2 = 3$ ,  $u_3 = 4$ ,  $u_4 = 4$ ,  $u_5 = 3$ ,  $u_6 = 5$ ,  $u_7 = 3$ , and  $u_8 = 6$ . A possible resource flow is shown in Figure 4.

Activity 8, for example, has a per period resource requirement of six units ( $u_8 = 6$ ). It uses three resource units released by its predecessor activity 5, two units passed on by activity 7 and one unit released by activity 6.

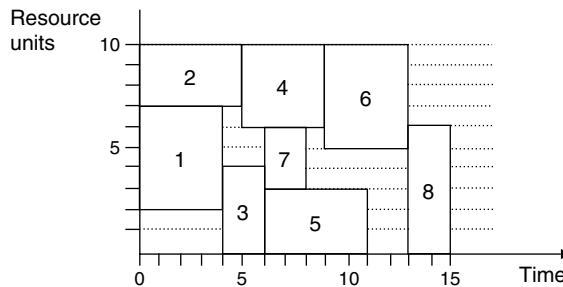
The set of arcs carrying nonzero flow,  $\Phi(f) = \{(0, 1), (0, 2), (0, 3), (1, 3), (1, 4), (1, 7), (2, 4), (3, 5), (3, 7), (4, 6), (5, 8), (6, 8), (6, 9), (7, 6), (7, 8), (8, 9)\}$  are shown in bold. The set of arcs  $\mathcal{X}(f) = \{(1, 3), (3, 7), (6, 8), (7, 6), (7, 8)\}$  represent extra precedence relations that were not present in the original project network. The arc  $(7, 9)$  does not carry flow.

Let  $\theta(X)$ ,  $X \subseteq N \times N$ , be the earliest start schedule in which the starting time of the dummy activity  $s_0 = 0$  and each other activity  $i$  starts at time  $s_i = \max_{j \in \pi_{A \cup X}(i)} \{s_j + p_j\}$ , with  $\pi_{A \cup X}(i)$  the immediate predecessors of activity  $i$  in the acyclic graph  $G(N, A \cup X)$ . A solution to an RCPSP instance can be obtained by finding a feasible flow  $f$  that minimizes  $s_n(\theta(A \cup \mathcal{X}(f)))$ .

The resource flows in Figure 4 may be represented in the resource profile shown in Figure 5, where the use of the 10 resource units is now shown along the 10 horizontal bands.

Leus and Herroelen [29] have shown that for any realizable flow  $f$ ,  $X = \mathcal{X}(f)$  defines a realizable *early start policy* (*ES-policy*). In order to obtain a feasible schedule for a given scenario of activity durations, an *ES-policy* simply computes earliest activity start times in a graph by performing a forward CPM (longest path) pass (Stork [36]). The idea behind an *ES-policy* (Igelmund and Radermacher [17]) is to extend the partially ordered set  $G(N, A)$  to

FIGURE 5. Resource profile showing the resource transfers.



a partially ordered set  $G(N, A \cup X)$  such that no so-called *forbidden sets*<sup>1</sup> remain precedence unrelated and can thereby not be scheduled in parallel. The feasibility condition for the policy is that  $G(N, A \cup X)$  still be acyclic. The arc (1, 3) in Figure 4, for example, guarantees that the three activities of the forbidden set  $\{1, 2, 3\}$  cannot be scheduled in parallel.

**A Branch-and-Bound Algorithm.** Leus and Herroelen [29] impose the restriction that a resource allocation be compatible with a precomputed baseline schedule  $\mathcal{S}^B$ . Let  $s_i(\mathcal{S}^B)$  denote the planned start time of activity  $i$  in the baseline schedule  $\mathcal{S}^B$  so that  $C_i(\mathcal{S}^B) = s_i(\mathcal{S}^B) + p_i$  denotes its corresponding planned completion time. Let  $\Lambda(\mathcal{S}^B) = \{(i, j) \in N \times N \mid (i, j) \notin TA, i \neq j, C_i(\mathcal{S}^B) \leq s_j(\mathcal{S}^B)\}$ . A feasible flow  $f$  is said to be compatible with a feasible baseline schedule  $\mathcal{S}^B$ , written  $f \sim \mathcal{S}^B$ , if  $\forall (i, j) \in TA \cup \mathcal{X}(f), C_i(\mathcal{S}^B) \leq s_j(\mathcal{S}^B)$ , or in other words  $\mathcal{X}(f) \subseteq \Lambda(\mathcal{S}^B)$ . One should attempt to respect the baseline schedule as much as possible: During project execution, activities are started at the maximum of the finish times of their predecessors and their baseline starting time (often referred to as *railway scheduling*). As such, the actual starting time of activity  $i$  is a stochastic variable  $S_i(P, \mathcal{X}(f), \mathcal{S}^B) = \max\{s_i(\mathcal{S}^B), \max_{j \in \pi_{TA \cup \mathcal{X}(f)}(i)} \{S_j(P, \mathcal{X}(f), \mathcal{S}^B) + P_j\}\}$ , with  $s_0(\mathcal{S}^B) = 0$ . Following the logic of Equation (6), the authors then aim at generating a feasible flow  $f$  with  $\mathcal{X}(f) \subseteq \Lambda(\mathcal{S}^B)$  such that

$$E \left[ \sum_{i \in N} w_i \times (S_i(P, \mathcal{X}(f), \mathcal{S}^B) - s_i(\mathcal{S}^B)) \right] \equiv g(\mathcal{X}(f)) \quad (9)$$

is minimized. Minimizing the expected makespan is the special case  $w_i = 0$ ,  $i \neq n$ , and  $w_n \neq 0$ .

The set of decision variables is the set of flows  $f_{ij}$  with  $(i, j) \in F = TA \cup \Lambda(\mathcal{S}^B)$ . For  $(i, j) \in F$ , denote by  $B_{ij}$  the domain initially associated with  $f_{ij}$ . The objective is to minimize the expression in Equation (9) subject to the constraints given by Equations (7)–(8) and the requirement that  $f \sim \mathcal{S}^B$ . Also,  $G_{\mathcal{X}(f)}$  is acyclic because arc  $(i, j) \in TA \cup \Lambda(\mathcal{S}^B)$  has  $C_i(\mathcal{S}^B) \leq s_j(\mathcal{S}^B) \leq C_j(\mathcal{S}^B)$  because the baseline schedule  $\mathcal{S}^B$  is feasible. For  $f_{ij} \in F$ ,  $B_{ij}$  can be represented by its minimal value  $LB_{ij}$  and its maximal value  $UB_{ij}$ . The domains are represented as intervals. The branch-and-bound algorithm implicitly evaluates all the valid flow values and relies on constraint propagation in order to reduce the search space. Leus and Herroelen [29] find an optimal resource allocation for a schedule  $\mathcal{S}^B$  by considering all subsets  $\Omega \subseteq \Lambda(\mathcal{S}^B)$  that allow a feasible flow in network  $TA \cup \Omega$ . One such subset corresponds with at least one and mostly multiple feasible  $f$ , with  $\mathcal{X}(f) \subseteq \Omega$ . They iteratively add arcs of  $\Lambda(\mathcal{S}^B)$  to  $\Omega$  until a feasible flow is attainable.

At each node  $k$  in the search tree the set  $F = TA \cup \Lambda(\mathcal{S}^B)$  is partitioned into three disjoint subsets:  $F = \alpha_k \cup \nu_k \cup \omega_k$ , with  $\alpha_k = \{(i, j) \in F, LB_{ij} > 0\}$  the set of included arcs,  $\nu_k = \{(i, j) \in F, UB_{ij} = 0\}$  the set of forbidden arcs, and  $\omega_k = \{(i, j) \in F, LB_{ij} = 0 \text{ and } UB_{ij} > 0\}$  the set of undecided arcs. Bounds  $LB_{ij}$  and  $UB_{ij}$  are established through constraint propagation in conjunction with branching conditions. All arcs in  $\alpha_k \setminus TA$  are added to  $\Omega_k$  which results in the partial network  $G_k = G_{\Omega_k}$ . If a feasible flow cannot be obtained in  $G_k$ , no further branching is needed, otherwise no further arcs must be added to the network and the procedure *backtracks* after having checked whether the objective function value corresponding with the current feasible solution improves on the objective function value of the best known feasible solution. The branching decision entails the selection of an undecided arc  $(i, j) \in \Lambda(\mathcal{S}^B) \cap \omega_k$ : the left branch is to set  $LB_{ij} = 1$ , so to include  $(i, j)$  in the partial network  $G_k$ . The right branch is to impose  $UB_{ij} = 0$ , so to forbid any flow across  $(i, j)$ , and prohibit inclusion of  $(i, j)$  in  $\Omega$  by placing the arc into set  $\nu_k$ . The result of the addition of a constraint is to split up the flow domain into two disjoint subsets, one of which is singleton  $\{0\}$ .

<sup>1</sup> A forbidden set is defined as a set of precedence unrelated activities, which cannot be scheduled together due to the resource constraints.

The authors report on promising computational results on randomly generated instances up to 61 activities (79% of the 61 activity instances are solved to optimality in an average CPU time of 45.5 seconds on a 800 MHz PC). Extension of the algorithm to multiple resource types would require a revision of the branching decisions taken by the branch-and-bound procedure and the consistency tests involved in the constraint propagation.

**Suboptimal Algorithms.** For the general *multiresource-type case*, Deblaere et al. [8] derive lower bounds on scheduling stability and develop and validate three integer programming-based resource-allocation heuristics and one constructive procedure against the flow generation algorithm of Artigues et al. [3] and three algorithms developed by Policella et al. [33]. Overall excellent results have been obtained using the constructive procedure MABO (myopic activity-based optimization). The procedure is myopic because the authors do not look at other activities while deciding on the best possible resource allocation for an activity. MABO consists of three steps which have to be executed for each activity  $j$ . Step 1 examines whether the current predecessors of activity  $j$  may release sufficient resource units to satisfy the resource requirements of activity  $j$ . If not, extra predecessors are added in Step 2 with a minimal impact on stability. Step 3 then defines resource flows  $f_{ijk}$  from predecessor activities  $i$  to activity  $j$  for renewable resource type  $k$ . The detailed steps of the procedure can be written as follows:

Initialize:  $A_R = A_U$  and  $\forall k: alloc_{0k} = a_k$

Sort the project activities by increasing  $s_j$  (tie break: decreasing  $w_j$ )

Take next activity  $j$  from list

1. Calculate  $Avail_{jk}(A \cup A_R) = \sum_{\forall i: (i,j) \in A \cup A_R} alloc_{ik}$  for each  $k$

2. If  $\exists k: Avail_{jk}(A \cup A_R) < r_{jk}$

2.1 Define the set of arcs  $H_j$

with  $(h, j) \in H_j \iff$

$(h, j) \notin A \cup A_R$

$s_h + p_h \leq s_j$

$\exists k: alloc_{hk} > 0$

2.2 Find a subset  $H_j^*$  of  $H_j$

such that  $\forall k: Avail_{jk}(A \cup A_R \cup H_j^*) \geq r_{jk}$

and  $Stability\_cost(A \cup A_R \cup H_j^*)$  is minimized

2.3 Add  $H_j^*$  to  $A_R$

3. Allocate resource flows  $f(i, j, k)$  to the arcs  $(i, j) \in (A \cup A_R)$ :

For each resource type  $k$ :

3.1 Sort predecessors  $i$  of  $j$  by:

Increasing number of successors  $l$  of  $i$

with  $s_l > s_j$  and  $r_{lk} > 0$

Tie break 1: Decreasing finish times  $s_i + p_i$

Tie break 2: Decreasing variance  $\sigma_i^2$  of  $P_i$

Exception: Activity 0 is always put last in the list

3.2 While  $alloc_{jk} < r_{jk}$

Take next activity  $i$  from the list

$f(i, j, k) = \min(alloc_{ik}, r_{jk} - alloc_{jk})$

Add  $f(i, j, k)$  to  $alloc_{jk}$

Subtract  $f(i, j, k)$  from  $alloc_{ik}$

Let us discuss the various steps of the algorithm in more detail. Let  $G = (N, A \cup A_R)$  denote the resource flow network, where  $A$  denotes the set of arcs representing the original precedence relations and the resource arcs  $A_R$  are connecting two nodes  $i$  and  $j$  if there is a resource flow  $f(i, j, k)$  of any resource type  $k$  between the corresponding activities  $i$  and  $j$ . In the initialization step, the set of resource arcs  $A_R$  is initialized to the set of unavoidable

arcs  $A_U \subset A_R$ . Two activities  $i$  and  $j$  must be connected by an *unavoidable resource arc* in the resource flow network for a given input schedule, if the schedule causes an unavoidable strictly positive amount of resource units  $f(i, j, k)$  of some resource type  $k$  to be sent from activity  $i$  to activity  $j$ . The conditions to be satisfied by activities  $i$  and  $j$  can be formally specified as follows:

$$\forall i \in N; \forall j \in N \text{ with } s_j \geq s_i + p_i: \\ (i, j) \in A_U \iff \exists k: a_k - \sum_{l \in \mathcal{A}_{s_j}} r_{lk} - \max\left(0, r_{ik} - \sum_{z \in Z} r_{zk}\right) < r_{jk} \quad (10)$$

in which  $\mathcal{A}_{s_j}$  is the set of the activities that are in progress at time  $s_j$  and  $Z$  is the set of activities that have a baseline starting time  $s_z$ :  $s_i + p_i \leq s_z < s_j$ . The left-hand side of Equation (10) identifies the number of resource units of type  $k$  that can be maximally supplied to activity  $j$  at time  $s_j$  from other activities than activity  $i$ . If this number is smaller than  $r_{jk}$ , there is an unavoidable resource flow between  $i$  and  $j$ . The exact amount and resource type of the flows on the unavoidable resource arc are irrelevant at this time. We are only interested in the fact that an arc  $(i, j)$  must be added to the set of unavoidable resource arcs  $A_U$ .

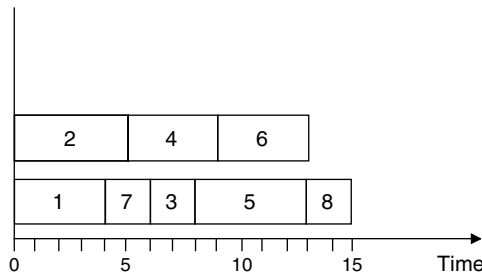
Figure 6 shows an alternative minimum duration schedule for the project example of Figure 1 (Van de Vonder [37]). This schedule requires an unavoidable resource arc from activity 7 to activity 3. At time  $s_3 = 6$  only activity 4 is in progress with  $r_4 = 4$ . Because activity 3 starts when activity 7 ends,  $Z$  is obviously void. This results in the left-hand side of Equation (10) being equal to  $10 - 4 - \max(0, 3 - 0) = 3$ , which is smaller than  $r_3 = 4$ . Arc  $(7, 3)$  should thus be added to  $A_U$ . The complete set of unavoidable arcs for the schedule in Figure 6 is  $A_U = \{(0, 1), (0, 2), (1, 7), (7, 3), (3, 5), (4, 6), (6, 8), (8, 9)\}$ .

For each resource type  $k$ , the number of resource units  $alloc_{0k}$  that may be transferred from the dummy start activity 0 is initialized to the resource availability  $a_k$ . The project activities are placed on a list in increasing order of their planned starting times using decreasing activity weight as tie-break rule.

Step 1 computes the amount of resource units  $Avail_{jk}(A \cup A_R)$  currently allocated to the predecessors of activity  $j$  in  $A \cup A_R$ .

If this amount of available resource units is not sufficient for any resource type  $k$ , new precedence constraints have to be added to  $A_R$  in Step 2. The set  $H_j$  is defined as the set of all possible arcs between a possible resource supplier  $h$  of the current activity  $j$  and  $j$  itself. By solving a small recursion problem, the subset  $H_j^*$  of  $H_j$  is found that accounts for the missing resource requirements of  $j$  for any resource type  $k$  at a minimum stability cost  $Stability\_cost(A \cup A_R \cup H_j^*)$ . The stability cost  $Stability\_cost(A \cup A_R \cup H_j^*)$  is the average stability cost  $\sum_{j \in N} w_j E|S_j - s_j|$ , computed through simulation of sufficient executions of the (partial) schedule, keeping the resource flows fixed, and respecting the additional precedence constraints  $A_R \cup H_j^*$  that were not present in the original project network diagram. The set

FIGURE 6. Alternative minimum duration schedule for the project of Figure 1.



of arcs  $H_j^*$  is added to  $A_R$  such that the updated  $Avail_{jk}(A \cup A_R) \geq r_{jk}$  and the resource-allocation problem for the current activity is solved in a myopic way.

In Step 3, the actual resource flows  $f(i, j, k)$  are allocated to the predecessors of  $j$  in  $A \cup A_R$  and  $alloc_{ik}$ , the number of resource items allocated to each activity, is updated. If  $Avail_{jk}(A \cup A_R) > r_{jk}$  for resource type  $k$ , it has to be decided which predecessors account for the resource flows. The predecessors  $i$  are sorted by increasing number of their not yet started successors  $l$  with  $r_{lk} > 0$ , because these successors might count on these resources to be available. Two tie-break rules are used: decreasing finish times and decreasing activity duration variances. The principle is that the predecessors earlier in the sorted list normally have a higher probability to disrupt future activities. It is advisable to consume all the resource units they release as much as possible such that their possible high impact on later activities is neutralized. This allocation procedure is redone for every resource type  $k$  independently. After all this, the three-step procedure is restarted for the next activity in the list until a complete feasible resource allocation is obtained at the end of the list. The procedure uses an optimal recursion algorithm for each activity, but is not necessarily optimal over all activities.

As an illustration, we run MABO on the minimum duration schedule of Figure 6. The activity weights are  $w_0 = 0$ ,  $w_1 = 2$ ,  $w_2 = 7$ ,  $w_3 = 4$ ,  $w_4 = 5$ ,  $w_5 = 3$ ,  $w_6 = 7$ ,  $w_7 = 5$ ,  $w_8 = 5$ , and  $w_9 = 38$ . Because the problem instance has a single resource type, we omit the index  $k$ . We start by ordering the activities, yielding the list  $(0, 2, 1, 7, 4, 3, 5, 6, 8, 9)$ . All available resource units are allocated to the dummy start activity ( $alloc_0 = 10$ ).

Activity 2 is next on the list. It has dummy activity 0 as single predecessor, so that  $Avail_2 = alloc_0 = 10$ . As  $Avail_2 > r_2$  ( $10 > 3$ ), no extra precedence relations have to be added and we can proceed to Step 3. We set  $f(0, 2) = \min(alloc_0, r_2 - alloc_2) = 3$ ,  $alloc_0 = 7$ , and  $alloc_2 = 3$ .

Also activity 1 poses no problems because its only predecessor (activity 0) still has 7 transferrable resource units and  $r_1 = 5$ . Thus,  $f(0, 1) = 5$ ,  $alloc_0 = 2$ ,  $alloc_1 = 5$ .

Activity 7 is the next activity on the list and we calculate in Step 1 that  $Avail_7((1, 7)) = alloc_1 = 5$  while  $r_7 = 3$ . Step 2 can thus again be skipped and the algorithm decides in Step 3 that  $f(1, 7) = \min(alloc_1, r_7 - alloc_7) = 3$ ,  $alloc_1 = 2$ , and  $alloc_7 = 3$ .

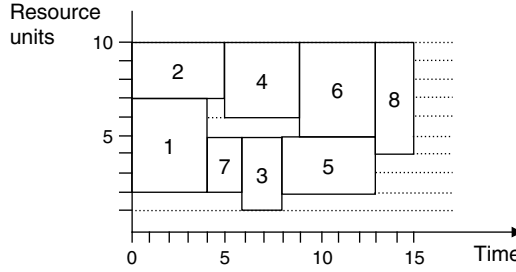
Activity 4 is next.  $Avail_4((1, 4), (2, 4)) = 2 + 3 = 5$  while  $r_4 = 4$ . Step 3 gives priority to activity 2, because neither activity 1 nor activity 2 has any not yet started predecessor left, but activity 2 ends later in the baseline schedule and is thus a greater stability threat for activities further down the list. This results in  $f(2, 4) = \min(alloc_2, r_4 - alloc_4) = 3$  and  $alloc_4 = 3$  and  $alloc_2 = 0$ . Then  $f(1, 4) = \min(alloc_1, r_4 - alloc_4) = 1$ ,  $alloc_1 = 1$ , and  $alloc_4 = 4$ . Activities 3 and 5 are processed in a similar way.

When activity 6 is looked at, the current situation is  $alloc_0 = 1$ ,  $alloc_1 = 1$ ,  $alloc_3 = 1$ ,  $alloc_4 = 4$ , and  $alloc_5 = 3$ , resulting in  $Avail_6(4, 6) = alloc_4 = 4$ , which is smaller than  $r_6 = 5$ . Thus, for the first time, an extra precedence relationship has to be added to supply one resource unit from  $H_6 = \{(0, 6), (1, 6), (3, 6)\}$ . Two subsets of  $H_6$ , namely  $(0, 6)$  and  $(1, 6)$  can resolve the resource-allocation problem for activity 6 without extra cost. This is no surprise because both 0 and 1 are already transitive predecessors of 6. Activity 1 is selected to supply the missing resource unit and thus  $(1, 6)$  is added to  $A_R$ . The procedure then moves on, until a complete feasible resource allocation is found.

Figure 7 shows the resource profile for the obtained robust resource allocation. The horizontal bands again define the resource flow between the activities.

**3.2.2. Buffer Insertion.** Van de Vonder [37] and Van de Vonder et al. [38]–[43] have developed several exact and suboptimal procedures for inserting time buffers in an input schedule under the objective of minimizing the stability cost function  $\sum_{i \in N} w_i E|S_i - s_i|$ , defined earlier in Equation (6). The included time buffers are idle periods (gaps) in the schedule that should act as cushions to prevent propagation of a disruption throughout the schedule.

FIGURE 7. MABO resource allocation for the schedule in Figure 6.



**Exact Algorithm.** Van de Vonder [37] has developed a depth-first branch-and-bound algorithm. The algorithm accepts as input an unbuffered schedule and accompanying resource allocation that will be preserved during schedule execution. An upper bound  $UB$  on the stability cost can be computed using a heuristic, for example the heuristic described in the next section. During the preprocessing phase, the activities are listed in decreasing order of their starting time  $s_j^U$  in the unbuffered input schedule  $\mathcal{S}^U$  (decreasing activity number as tie break). For the input schedule of Figure 2, this would yield the list  $L = (l_{[1]}, l_{[2]}, \dots, l_{[n]}) = (9, 8, 6, 7, 5, 4, 3, 2, 1, 0)$ . The set  $N_{j-}$  denotes the set of activities preceding activity  $j$  in the list, whereas the set  $N_{j+}$  denotes the set of its successor activities in the list. Let  $es_j = s_j^U$  be the earliest allowable start time of activity  $j$ . The project due date  $d_n$  defines the latest allowable start time  $ls_n$  for the dummy end activity  $n$ . Backward calculations in the network  $G = (N, A \cup R)$  then yield the latest allowable start time  $ls_j$  for each activity  $j$ . The total schedule float of activity  $j$ , the maximum amount of time by which activity  $j$  may be delayed without violating the due date  $d_n$ , is then computed as  $TF_j = ls_j - es_j$ . A lower bound  $stab_j^{\min}(FF_j)$  on the stability cost induced by activity  $j$  when activity  $j$  is preceded by a time buffer of  $FF_j$  time units, is then computed by running the following algorithm:

$\forall j \in N$ :  
**for**  $FF_j = 0$  **to**  $(ls_j - es_j)$  **do**  
    Simulate sufficient scenarios of  $G(N_{j+}, A \cup A_R)$  with:  
         $\forall i \in N_{j+}: s_i = es_i$   
         $s_j = es_j + FF_j$   
    Calculate  $stab_j^{\min}(FF_j) = w_j E|S_j - s_j|$ .

The depth first search considers the activities in the order dictated by the list  $L = (l_{[1]}, l_{[2]}, \dots, l_{[n]})$ . The activity under consideration is called the *current activity*. The first activity  $l_{[1]}$  on the list is the dummy end activity  $n$ . The root node at level 0 of the search tree is generated with  $s'_n = d_n$ . Moving to the second current activity  $c = l_{[2]}$  on the list, the left most node at level 1 of the search tree is generated with  $s'_c = s_c$ . The lower bound on the stability cost induced by this current activity  $c$  is computed using simulation as

$$LB1_c = \sum_{l \in N_{c-}} w_l E|S_l - s_l|. \quad (11)$$

If  $LB1_c \geq UB$ , then the node can be fathomed, as it is not necessary to evaluate the starting times  $s_i > es_i$  for any  $i \in N_{c+}$ , given that  $s'_l$  has been fixed for each  $l \in N_{c-}$ . Also, it is not necessary to evaluate other starting times for  $c$ . Hence, the procedure *backtracks* to the previous activity in the list.

Backtracking is done by moving one level up in the search tree and investigating the next position  $s'_{(c')} + 1$  for the current activity  $c'$  explored at that level. If there exists no subsequent starting time of  $c'$  such that  $s_{c'} + p_{c'} \leq \max_{\forall m: (c', m) \in (A \cup A_R)} s'(m)$ , then backtracking continues until an activity is met with feasible subsequent starting times. When backtracking reaches the root of the search tree, the algorithm stops.

If  $LB1_c < UB$ , a tighter lower bound  $LB2_c$  can be computed. Latest starting times  $ls'_i$  for  $i \in N_{c+}$  are calculated given that  $\forall l \in (N_{c-} \cup \{c\})$ :  $ls'_l = s'_l$ . For any activity  $i \in (N_{c+} \cup \{c\})$ , the expression  $w_i E|S_i - s_i| \geq stab_i^{\min}(ls'_i - es_i)$  holds for any schedule that would be generated in lower branches of the search tree. Aggregation yields

$$\sum_{i \in (N_{c+} \cup \{c\})} w_i E|S_i - s_i| \geq \sum_{i \in (N_{c+} \cup \{c\})} stab_i^{\min}(ls'_i - es_i). \quad (12)$$

We have shown above that  $\sum_{i \in N_{c-}} w_i E|S_i - s_i| \geq LB1$  holds in the current branch of the search tree. It results that

$$LB2 = LB1 + \sum_{i \in (N_{c+} \cup \{c\})} stab_i^{\min}(ls'_i - es_i) \leq \sum_{i \in N} w_i E|S_i - s_i|. \quad (13)$$

$LB2$  is thus a lower bound on the stability cost for all schedules with  $s_j = s'_j$  and  $\forall l \in N_{j-}$ :  $s_l = s'_l$ . The computations made in the preprocessing stage make this bound rather easy to compute. If  $LB2 \geq UB$ , the current node can be fathomed and the search continues by evaluating the next larger start time for current activity  $c$ . If  $LB2 < UB$ , the current branch needs to be further explored by branching on the next activity in the ordered list. When branching has been done for all activities  $i$  with  $es_i > 0$  and  $LB2 < UB$ , a new best solution has been found with stability cost  $LB2$ .

Running the optimal buffer insertion algorithm on the input schedule given in Figure 2 with  $d_n = 20$ , activity weights  $w_0 = 0$ ,  $w_1 = 2$ ,  $w_2 = 7$ ,  $w_3 = 4$ ,  $w_4 = 5$ ,  $w_5 = 3$ ,  $w_6 = 7$ ,  $w_7 = 5$ ,  $w_8 = 5$ , and  $w_9 = 38$ , and expected activity durations  $E(D_1) = 4$ ,  $E(D_2) = 5$ ,  $E(D_3) = 2$ ,  $E(D_4) = 4$ ,  $E(D_5) = 5$ ,  $E(D_6) = 4$ ,  $E(D_7) = 2$ , and  $E(D_8) = 2$ , yields the robust project schedule of Figure 8.

**A Heuristic Procedure.** Several heuristic procedures have been developed for generating stable buffered schedules under the  $\sum_{i \in N} w_i E|S_i - s_i|$  objective (Van de Vonder et al. [39]). Despite the simplicity of its underlying assumptions and structure, the *starting time criticality heuristic* (STC) obtained excellent results.

The STC exploits information about both the weights of the activities and the variance structure of the activity durations. The basic idea is to start from a minimum duration schedule and iteratively create intermediate schedules by adding a one-unit time buffer in front of that activity that needs it the most in the current intermediate schedule, until adding more safety would no longer improve stability. We thus need a measure to quantify for each activity how critical its current starting time is in the current intermediate baseline schedule. The *starting time criticality* for activity  $j$  is defined as  $stc(j) = P(S_j > s_j) \times w_j = \gamma_j \times w_j$ , where  $\gamma_j$  denotes the probability that activity  $j$  cannot be started at its scheduled starting time  $s_j$ .

The iterative procedure runs as follows. At each iteration step (see Figure 9), the buffer sizes of the current intermediate schedule are updated as follows. The activities are listed

FIGURE 8. Buffered schedule produced by branch-and-bound.

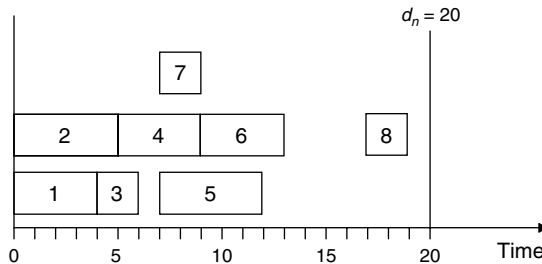




FIGURE 9. Iteration step of STC heuristic.

```

Calculate all  $stc(j)$ 
Sort activities in decreasing order of the  $stc(j)$ 
While no improvement found do
  Take next activity  $j$  from list
  If  $stc(j) := 0$  then procedure terminates
  Else add buffer in front of  $j$ 
    Update schedule
    If improvement and feasible then
      Store schedule
      Goto next iteration step
    Else
      Remove buffer in front of  $j$ 
      Restore schedule

```

in decreasing order of the  $stc(j)$ . The list is scanned and the size of the buffer to be placed in front of the currently selected activity from the list is augmented by one time unit such that the starting times of the activity itself and of the direct and transitive successors of the activity in  $G(N, A \cup R)$  are increased by one time unit. If this new schedule has a feasible project completion ( $s_n < d_n$ ) and results in a lower approximated stability cost ( $\sum_{j \in N} stc(j)$ ), the schedule serves as the input schedule for the next iteration step. If not, the next activity in the list is considered. Whenever an activity  $j$  is encountered for which  $stc(j) = 0$  (all activities  $j$  with  $s_j = 0$  are by definition in this case) and no feasible improvement is found, a local optimum is obtained and the procedure terminates.

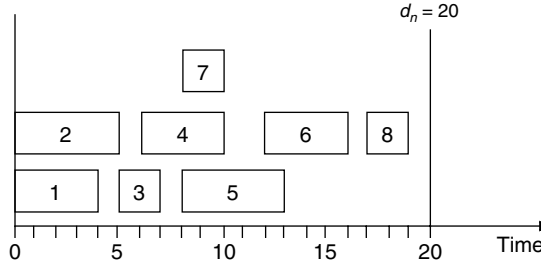
Regrettably, the probabilities  $\gamma_j$  are not easy to compute. The authors define  $k(i, j)$  as the event that predecessor  $i$  disturbs the planned starting time of activity  $j$ . The probability that this event occurs can be expressed as  $P(k(i, j)) = P(S_i + P_i + LPL(i, j)) > s_j$ , in which  $LPL(i, j)$  is the sum of the durations of all activities  $h$  on the longest path between activity  $i$  and activity  $j$  in the graph  $G(N, A \cup R)$ . We can now calculate  $\gamma_j = P(\bigcup_i k(i, j))$  for  $\forall i: (i, j) \in T(A \cup R)$ .

STC makes two assumptions in approximating  $\gamma_j$ : (a) only one activity at a time disturbs the starting time of activity  $j$ , and (b) the predecessor  $i$  of activity  $j$  starts at its originally planned starting time. Assumption (a) means that  $P(\bigcup_i k(i, j))$  is estimated by  $\sum_i P(k(i, j))$ , i.e., it is assumed that  $P(k(i1, j) \cap k(i2, j)) = 0$  for each  $i1, i2$ . Assumption (b) boils down to setting  $S_i = s_i$ . Combining both assumptions yields  $\gamma'_j = \sum_i P(P_i > s_j - s_i - LPL(i, j))$  such that  $stc(j) = \gamma'_j \times w_j$ . Because  $s_i, s_j, LPL(i, j)$  and the distribution of the  $P_i$  are all known, the values of  $\gamma'_j$  and  $stc(j)$  can be easily computed for every activity  $j$ .

Assuming activity weights  $w_0 = 0, w_1 = 2, w_2 = 7, w_3 = 4, w_4 = 5, w_5 = 3, w_6 = 7, w_7 = 5, w_8 = 5$ , and  $w_9 = 38$ , and expected activity durations  $E(D_1) = 4, E(D_2) = 5, E(D_3) = 2, E(D_4) = 4, E(D_5) = 5, E(D_6) = 4, E(D_7) = 2$ , and  $E(D_8) = 2$ , the application of the STC algorithm to the input schedule of Figure 2 runs as follows. We illustrate the calculation of the  $LPL(i, j)$  for  $LPL(1, 3)$ ,  $LPL(1, 5)$ , and  $LPL(1, 8)$ . Activity 3 is immediately preceded by activity 1 in the schedule, hence  $LPL(1, 3) = 0$ . The resource flow network of Figure 4 shows a unique path  $\langle 1, 3, 5 \rangle$  leading from activity 1 to activity 5, yielding  $LPL(1, 5) = E(D_3) = 2$ . Multiple paths exist between activity 1 and activity 8, namely  $\langle 1, 3, 5, 8 \rangle$ ,  $\langle 1, 3, 7, 8 \rangle$ ,  $\langle 1, 3, 7, 6, 8 \rangle$ ,  $\langle 1, 7, 8 \rangle$ ,  $\langle 1, 7, 6, 8 \rangle$ , and  $\langle 1, 4, 6, 8 \rangle$ , with corresponding path length of 7, 4, 8, 4, 2, 6, and 8 respectively. Thus,  $LPL(1, 8) = 8$ .

The  $stc$ -values are first calculated for the initial minimum duration schedule of Figure 2 with due date  $d_n = 20$ . For example,  $stc(6)$  is calculated as  $w_6 \times (P(k(1, 6)) + P(k(2, 6)) + P(k(3, 6)) + P(k(4, 6)) + P(k(7, 6)))$ , with  $P(k(1, 6)) = P(P_1 > s_6 - s_1 - LPL(1, 6)) = P(P_1 > 5) = 0.11$ ,  $P(k(2, 6)) = P(P_2 > s_6 - s_2 - LPL(2, 6)) = P(P_2 > 5) = 0.23$ ,  $P(k(3, 6)) = P(P_3 > s_6 - s_3 - LPL(3, 6)) = P(P_3 > 3) = 0.01$ ,  $P(k(4, 6)) = P(P_4 > s_6 - s_4 - LPL(4, 6)) = P(P_4 > 4) = 0.34$ , and  $P(k(7, 6)) = P(P_7 > s_6 - s_7 - LPL(7, 6)) = P(P_7 > 5) = 0.05$ . As a

FIGURE 10. STC schedule for the example problem.



result,  $stc(6) = 7 \times (0.11 + 0.23 + 0.01 + 0.34 + 0.05) = 5.18$ . Computing the other  $stc$ -values yields  $\sum stc(j) = 0 + 0 + 1.20 + 1.71 + 1.47 + 5.18 + 2.45 + 4.88 + 0.04 = 16.93$  as the total schedule cost. Ordering the activities by decreasing  $stc$  gives the list  $(6, 8, 7, 4, 5, 3, 9, 1, 2)$ . Adding a one-time period buffer in front of activity 6 yields updated start times  $s_6 = 10$  and  $s_8 = 14$ . The newly inserted buffer in front of activity 6 requires a recalculation of its  $stc$ -value and the  $stc$ -values of its successor activities 8 and 9. Activity 7 now has the largest  $stc$ -value, yielding the ordered list  $(7, 8, 4, 6, 5, 3, 9, 1, 2, 0)$ . Delaying the starting time of activity 7 is feasible and leads to a reduction in the total schedule cost  $\sum stc(j) = 0 + 0 + 1.20 + 1.71 + 1.47 + 1.69 + 2.45 + 2.22 + 0.04 = 10.78$ . Subsequently, the procedure will insert a time buffer in front of activity 8, activity 6, activity 4 (leading to a one-period delay in activity 6 and 8), activity 5, activity 3 (leading to a one-period delay of activities 5 and 7). The procedure then continues by examining a possible delay of activity 6. Delaying activity 6, however, would lead to an increase in the total schedule cost. Therefore, activity 6 is not delayed. In a similar fashion, delaying activity 3 or either of the following activities in the list would yield no cost improvement. The procedure terminates with the schedule shown in Figure 10 (Van de Vonder [37]).

### 3.3. Reactive Scheduling

Proactive/reactive scheduling implies that the buffered baseline schedules generated by the proactive procedures should be combined with reactive scheduling procedures that are deployed during project execution when disruptions occur that cannot be absorbed by the baseline schedule.

The literature concerning reactive project scheduling is virtually void. Yu and Qi [48] describe an ILP model for the multimode RCPSP (problem  $m, 1T|cpm, disc, mu|C_{\max}$  in the notation of Herroelen et al. [15]) and report on computational results obtained by a hybrid mixed integer programming/constraint propagation approach for minimizing the schedule deviation caused by a single disruption induced by a known increase in the duration of a single activity. Van de Vonder et al. [40] developed and extensively evaluated a number of exact and heuristic reactive procedures.

The reactive scheduling problem at decision point  $t$  when the baseline schedule  $\mathcal{S}^B$  breaks, can be viewed as a resource-constrained project scheduling problem with weighted earliness tardiness costs (problem  $m, 1|cpm|early/tardy$  in the notation of Herroelen et al. [15]). Due dates are set equal to the activity completion times  $s_i + p_i$  in the predictive schedule. The earliness and tardiness costs may be assumed to be symmetrical and chosen as the weights  $w_i$  in the stability objective function, with a possible exception for the earliness cost of the dummy end activity, which can be set equal to zero.

Efficient exact procedures for solving problem  $m, 1|cpm|early/tardy$  have been proposed in the scheduling literature (see e.g., Vanhoucke et al. [44] and Kéri and Kis [18]). However, Van de Vonder et al. [40] found that calling an exact weighted earliness-tardiness procedure at each schedule breakage point becomes computationally infeasible already for small networks. The authors obtained excellent computational results with a sampling approach.

The basic *sampling approach* by Van de Vonder et al. [40] relies on different priority lists in combination with different schedule generation schemes. It tries to make a suitable decision at each decision time  $t$  as follows ( $\mathcal{S}^0$  is the baseline schedule):

```

for  $t = 0, \dots, T$  do
  Step 1: Check for new scheduling information.
  Step 2: If no new information then  $\mathcal{S}^t = \mathcal{S}^{t-1}$  and goto period  $t + 1$ 
           else goto step 3
  Step 3: For list  $\lambda_l = \lambda_0, \dots, \lambda_L$  do
           Construct  $\mathcal{S}_{\lambda_l, RP}^t$  and calculate  $\Delta(\mathcal{S}^0, \mathcal{S}_{\lambda_l, RP}^t)$ 
           Construct  $\mathcal{S}_{\lambda_l, RS}^t$  and calculate  $\Delta(\mathcal{S}^0, \mathcal{S}_{\lambda_l, RS}^t)$ 
           Construct  $\mathcal{S}_{\lambda_l, P}^t$  and calculate  $\Delta(\mathcal{S}^0, \mathcal{S}_{\lambda_l, P}^t)$ 
           Construct  $\mathcal{S}_{\lambda_l, S}^t$  and calculate  $\Delta(\mathcal{S}^0, \mathcal{S}_{\lambda_l, S}^t)$ 
           Store the schedule  $\mathcal{S}^t$  that minimizes  $\Delta(\mathcal{S}^0, \mathcal{S}^t)$ 
  Step 4: Start all activities  $i$  with  $s_i^t = t$ .

```

Step 1 checks for new information becoming available at time  $t$ . If at time  $t$ , no activity finishes and no activity was projected to finish ( $s_i^{t-1} = t$ ), then no new information became available since the previous decision point  $t - 1$ . The previous projected schedule  $\mathcal{S}^{t-1}$ , generated at time  $t - 1$ , remains valid (Step 2).

Instead of using one priority list in combination with one schedule generation scheme (SGS), Step 3 uses multiple lists  $\lambda_l = \lambda_0, \dots, \lambda_L$  at time  $t$  in combination with several SGSs.

The authors evaluate different priority lists: EBST (earliest start time in the baseline schedule), LST (latest starting time), LW (largest activity weight), LAN (lowest activity number), RND (random), EPST (earliest starting time in the schedule generated at the last decision point), and MC (lowest current stability cost).

For each of these priority lists  $\lambda_l$ , a complete schedule is generated using four schedule generation schemes. The *parallel schedule generation scheme* ( $\mathcal{S}_{\lambda_l, P}^t$ ) iterates over time and starts at each decision point, in the order dictated by the priority list, as many unscheduled activities as possible in accordance with the precedence and resource constraints. The *robust parallel schedule generation scheme* ( $\mathcal{S}_{\lambda_l, RP}^t$ ) is similar to the parallel scheme, but considers at each decision time  $t$  only the activities for which the current decision time  $t$  is greater than or equal to their planned starting time in the baseline schedule. The *serial schedule generation scheme* ( $\mathcal{S}_{\lambda_l, S}^t$ ) schedules at each decision point  $t$  the next activity from the priority list. The *robust serial schedule generation scheme* ( $\mathcal{S}_{\lambda_l, RS}^t$ ) considers the activities in the order dictated by the priority list and starts them at a feasible time as close as possible to their planned starting time in the baseline schedule.

In this way, a total of  $4 \times L$  candidate schedules are generated and the schedule  $\mathcal{S}_{\lambda_l, .}^t$ , yielding the smallest stability cost deviation  $\Delta(\mathcal{S}^0, \mathcal{S}_{\lambda_l, .}^t)$  from the baseline schedule  $\mathcal{S}^0$  is stored. The procedure then continues in Step 4 by starting the activities for which the planned starting time in the schedule equals  $t$ .

## 4. Solution Robust Scheduling Under Resource Availability Uncertainty

The literature on proactive/reactive project scheduling under resource availability uncertainties is virtually void. Drezet [10] considers the problem of project planning subject to human-resource constraints, which have to do with job competences, working hour limits, vacation periods, and unavailability of employees. She presents a mathematical model as well as dedicated algorithms for robust schedule generation and schedule repair. Yu and Qi [48], in their above-mentioned ILP model for the multimode problem, allow for (known) decreases in the resource availabilities in certain planning periods.

## 4.1. The Problem

Contrary to §3, activity durations are now assumed to be deterministic; uncertainty originates from the stochastic nature of renewable resource availability  $A_k$  ( $k = 1, \dots, q$ ). This means that during schedule execution infeasibilities may occur due to renewable resource breakdowns, so that the schedule needs to be repaired. The proactive project scheduling problem then consists of generating a proactive schedule that is as well as possible protected from such disruptions, subject to a project deadline, finish-start, zero-lag precedence constraints and renewable resource constraints.

A maximum resource availability  $a_k$  is considered for each renewable resource type  $k$ . Each of these  $a_k$  resource units initially allocated to the project is subject to breakdowns; some of the proposed models presuppose knowledge of the mean time to failure and the mean time to repair. The objective function to be minimized is again  $\sum_{i \in N} w_i E(|S_i - s_i|)$ , the weighted expected deviation between the planned and actually realized activity start times.

Lambrechts et al. [22] develop and evaluate eight proactive and three reactive scheduling procedures. A tabu search procedure for generating robust baseline schedules is presented in Lambrechts et al. [23]. In Lambrechts et al. [24], the authors analytically determine the impact of unexpected resource breakdowns on activity durations and develop effective and efficient algorithms for inserting explicit idle time into an initial unbuffered input schedule. Reactive strategies are discussed in Lambrechts et al. [25].

**4.1.1. Proactive Strategies.** The two-step proactive scheduling procedure (Lambrechts et al. [24]) for generating stable baseline schedules may take as input an unbuffered schedule generated by an exact or heuristic procedure for solving the deterministic RCPSP, or by a procedure that places activities that can be expected to have a high impact on the total project stability earliest in time (*largest CIW first*). Furthermore, it can be decided to include resource slack (*resource buffering*).

Resource buffering boils down to planning the project subject to a deterministic nominal resource availability that is strictly below the maximum deterministic resource availability  $a_k$ . More precisely, the nominal availabilities are set equal to  $E(A_k) = \sum_{m=0}^{a_k} (a_k - m) \pi_m$ , where  $\pi_m$  denotes the steady state probability that  $m$  resource units of resource type  $k$  are inactive. When necessary, this value is increased to  $\max_{i \in N} r_{ik}$  to allow for the activity with the highest resource demand for resource type  $k$  to be scheduled.

The largest-CIW-first rule schedules the activities  $i$  in nonincreasing order of their cumulative instability weight  $CIW_i = w_i + \sum_{j \in Succ_i} w_j$ , where  $Succ_i$  denotes the set of direct and indirect successors of activity  $i$ . In a first step a precedence feasible priority list is constructed in which precedence-unrelated activity pairs appear in nonincreasing order of  $CIW_i$  (lowest activity number as tie breaker). Afterwards this priority list is transformed into a precedence and resource feasible schedule using a serial schedule generation scheme that sequentially adds activities to the schedule until a feasible complete schedule is obtained. In each step, the next activity in the priority list is selected and for that activity the first precedence and resource feasible starting time is chosen.

Time buffers can then be inserted into the unbuffered schedule in order to increase its stability. *Time buffering* implies that time buffers are inserted in front of activities in order to absorb potential disruptions caused by earlier resource breakdowns and the resulting activity shifts. The input schedule may be iteratively buffered using a simulation-based steepest descent procedure (Lambrechts et al. [24]). In each iteration, every activity (except the dummy start) is considered for buffering. The selected activity is then right-shifted with one time unit. Affected activities are likewise right-shifted with one time unit in order to keep the schedule precedence and resource feasible. The activity leading to the greatest stability cost reduction (determined by simulation) that yields a schedule respecting the deadline is buffered. If no such activity can be found, the procedure terminates.

Such a simulation-based procedure is very time consuming. Surrogate measures can be used to estimate the instability costs. Lambrechts et al. [24] conclude that for the preempt-repeat case, in which an interrupted activity must be restarted later, the best results are obtained by computing the surrogate measure as  $\sum_{j \in N} \sum_{i \in Pred_j} w_j \max(0, s_i + p_i + LPL_{ij} + E[\sigma_i] - s_j)$ , where  $Pred_j$  denotes the immediate and transitive predecessors of activity  $j$ ,  $LPL_j$  represents the length of the longest path between activities  $i$  and  $j$  in  $G(N, A \cup R)$ , and  $E[\sigma_i]$  denotes the expected duration extension of activity  $i$  caused by the resource breakdown.

Lambrechts et al. [24] conclude from their computational experiment, assuming exponential or uniform repair time distributions, and combining the proactive procedure with the scheduled order reactive procedure (§4.1.2), that simulation-based time buffering always outperforms the time-buffering approaches that use surrogate stability cost estimates. However, its computational requirements are prohibitive. In the preempt-resume case, when interrupted activities may be resumed on repair of the broken resource(s), and in the preempt-setup case, when a setup time is needed when activities are resumed, best results are obtained using the STC heuristic described earlier. In the absence of resource and time buffering, the largest CIW scheduling rule outperforms the use of a minimum makespan input schedule. Resource buffering always pays off.

**4.1.2. Reactive Strategies.** When the baseline schedule breaks, i.e., when activities have to be interrupted because of a resource breakdown, the schedule needs to be repaired using a reactive procedure. Lambrechts et al. [22] investigate a preempt-repeat setting (interrupted activities have to be restarted anew); they generate a list  $L$  containing all activities that are not yet completed at the time of interruption, ordered in nondecreasing order of the baseline starting times. This list is then decoded into a feasible schedule using a serial schedule generation scheme that tries to schedule the interrupted activities as early as possible; a tabu search algorithm to improve the generated reactive schedule is also proposed in the same source.

Lambrechts et al. [25] study exact and suboptimal procedures to restore schedule feasibility under the objective of minimizing the weighted sum of deviations between the repaired schedule and the baseline schedule, under the assumption that the encountered disruption is the last disruption until project completion. The exact algorithm relies on the (truncated) branch-and-bound algorithm of Vanhoucke et al. [44] for solving the resulting resource-constrained project scheduling problem with weighted earliness tardiness costs (problem  $m, 1|cpm|early/tardy$  in the notation of Herroelen et al. [15]). They also present a *scheduled order list scheduling heuristic* that allows to reschedule the activities in the order dictated by the baseline schedule (the lowest activity number being the tie breaker) although taking into account the new, reduced resource availabilities. They obtain improved solutions by imposing a tabu search procedure on the priority list rule.

Lambrechts et al. [25] extend the tabu search procedure allowing it, when a disruption occurs, not only to generate a repaired baseline schedule that does not deviate too much from the original baseline, but a repaired schedule that is also protected against the occurrence of future disruptions. They use a surrogate robustness measure based on the expected duration increase of an activity due to resource breakdowns.

## 5. Conclusions

Real-life projects are typically subject to considerable uncertainty. This chapter has addressed proactive/reactive project scheduling procedures that may be deployed when the uncertainty pertains to the duration of activities or to the availability of renewable resources. *Proactive* procedures have been described to generate a robust baseline schedule that is appropriately protected against distortions that may occur during project execution. The term “robustness” in this context refers to solution robustness or stability. Our aim has

been to generate proactive precedence and resource feasible baseline schedules that minimize one particular stability cost function, namely the weighted sum of the expected deviation between the actually realized activity start times during project execution and the planned activity start times in the baseline. When distortions during project execution cause the baseline schedule to become infeasible, a *reactive* policy needs to be invoked to repair the schedule.

For variable activity durations, Van de Vonder et al. [37–43] have developed exact and heuristic proactive time-buffer-insertion strategies that can be combined with effective reactive policies for (optimally or heuristically) solving the underlying weighted earliness-tardiness problem. These research efforts allow to draw interesting and reassuring conclusions. It appears that the combination of proactive and reactive scheduling techniques leads to significant stability improvements (reduction in the planning nervousness), with only moderate (hence acceptable) increases in schedule makespan.

The unavailability of resources is a second potential but very realistic cause of substantial deviations from the baseline schedule. Consequently, the development of proactive/reactive scheduling procedures under stochastic resource availability is relevant from a theoretical and a practical point of view. Research in this area is just emerging. Lambrechts et al. [22–25] have obtained excellent results with their proactive/reactive procedures to cope with resource breakdowns.

These promising results justify the engagement in additional research. The development of effective and efficient single-step (monolithic) proactive scheduling procedures for the generation of stable (solution robust) baseline schedules with acceptable makespan performance, which can be easily combined with effective reactive scheduling policies that are able to operate under various types of schedule distortions, deserves priority. The exploration of robustness measures other than the weighted activity starting time deviations, which was explored in this chapter, constitutes another interesting area of future research.

## Acknowledgments

This research has been supported by Project OT/03/14 of the Research Fund of Katholieke Universiteit Leuven, Project G.0109.04 of the Research Foundation – Flanders (FWO-Vlaanderen) and Project NB06163 supported by the National Bank of Belgium. The author is very much indebted to Filip Deblaere, Erik Demeulemeester, Olivier Lambrechts, Roel Leus, and Stijn Van de Vonder. Their research results, obtained over the last few years, were indispensable for the preparation of this tutorial.

## References

- [1] M. Aloulou and M.-C. Portmann. An efficient proactive scheduling approach to hedge against shop floor disturbances. *Proceedings of the First Multidisciplinary Conference on Scheduling: Theory and Applications*, Nancy, France, 337–362, 2003.
- [2] C. Artigues and F. Roubellat. A polynomial activity insertion algorithm in a multi-resource schedule with cumulative constraints and multiple modes. *European Journal of Operational Research* 127(2):297–316, 2000.
- [3] C. Artigues, P. Michelon, and S. Reusser. Insertion techniques for static and dynamic resource-constrained project scheduling. *European Journal of Operational Research* 149(2):249–267, 2003.
- [4] H. Aytug, M. A. Lawley, K. McKay, S. Moan, and R. Uzsoy. Executing production schedules in the face of uncertainties: A review and some future directions. *European Journal of Operational Research* 161:86–110, 2005.
- [5] J.-C. Billaut, A. Moukrim, and E. Sanlaville. *Flexibilité et robustesse en ordonnancement. Traité IC2, Série Informatique et systèmes d'information*. Hermes Science Publications, Paris, France, 2005.
- [6] J. Blazewicz, J. K. Lenstra, and A. H. G. Rinnooy Kan. Scheduling subject to resource constraints—classification and complexity. *Discrete Applied Mathematics* 5(1):11–24, 1983.

- [7] P. Brucker, A. Drexler, R. Möhring, K. Neumann, and E. Pesch. Resource-constrained project scheduling: Notation, classification, models and methods. *European Journal of Operational Research* 112(1):3–41, 1999.
- [8] F. Deblaere, E. Demeulemeester, W. Herroelen, and S. Van de Vonder. Robust resource-allocation decisions in resource-constrained projects. *Decision Sciences* 38(1):5–37, 2007.
- [9] E. Demeulemeester and W. Herroelen. *Project Scheduling—A Research Handbook. International Series in Operations Research & Management Science*, Vol. 49. Springer, Heidelberg, Germany, 2002.
- [10] L.-E. Drezet. Résolution d'un problème de gestion de projets sous contraintes de ressources humaines: De l'approche prédictive à l'approche réactive. Ph.D. thesis, Université François Rabelais, Tours, France, 2005.
- [11] J. N. Hagstrom. Computational complexity of PERT problems. *Networks* 18(2):139–147, 1988.
- [12] W. Herroelen and R. Leus. Robust and reactive project scheduling: A review and classification of procedures. *International Journal of Production Research* 42(8):1599–1620, 2004.
- [13] W. Herroelen and R. Leus. Project scheduling under uncertainty—Survey and research potentials. *European Journal of Operational Research* 165(2):289–306, 2005.
- [14] W. Herroelen, B. De Reyck, and E. Demeulemeester. Resource-constrained scheduling: A survey of recent developments. *Computers and Operations Research* 25(4):279–302, 1998.
- [15] W. Herroelen, B. De Reyck, and E. Demeulemeester. On the paper “Resource-constrained project scheduling: Notation, classification, models and methods” by Brucker et al. *European Journal of Operational Research* 128(3):679–688, 2001.
- [16] H. Hoogeveen. Multicriteria scheduling. *European Journal of Operational Research* 167(3):592–623, 2004.
- [17] G. Igelmund and F. J. Radermacher. Algorithmic approaches to preselective strategies for stochastic scheduling problems. *Networks* 13(1):29–48, 1983.
- [18] A. Kéri and T. Kis. Primal-dual combined with constraint propagation for solving RCPSP-WET. *Proceedings of the 2nd Multidisciplinary International Conference on Scheduling: Theory and Applications*, MISTA, New York. 748–751, 2005.
- [19] N. Kinnock. Communication of the European Commission to the Council concerning the Berlaymont Building. 1094, 2002.
- [20] R. Kolisch and R. Padman. An integrated survey of deterministic project scheduling. *Omega* 49(3):249–272, 1999.
- [21] P. Kouvelis and G. Yu. *Robust Discrete Optimization and Its Applications*. Kluwer Academic Publishers, Boston, MA, 1997.
- [22] O. Lambrechts, E. Demeulemeester, and W. Herroelen. Proactive and reactive strategies for resource-constrained project scheduling with uncertain resource availabilities. *Journal of Scheduling*. Forthcoming.
- [23] O. Lambrechts, E. Demeulemeester, and W. Herroelen. A tabu search procedure for developing robust predictive project schedules. *International Journal of Production Economics*. Forthcoming.
- [24] O. Lambrechts, E. Demeulemeester, and W. Herroelen. Time-slack-based techniques for generating robust project schedules subject to resource uncertainty. Research report KBI, Department of Decision Sciences and Information Management (KBI), Katholieke Universiteit Leuven, Leuven, Belgium, 2007.
- [25] O. Lambrechts, E. Demeulemeester, and W. Herroelen. Exact and suboptimal reactive strategies for resource-constrained project scheduling with uncertain resource availabilities. Research report KBI 0702, Department of Decision Sciences and Information Management (KBI), Katholieke Universiteit Leuven, Leuven, Belgium, 2007.
- [26] V. J. Leon, S. D. Wu, and R. H. Storer. Robustness measures and robust scheduling for job shops. *IIE Transactions* 16(5):32–43, 1994.
- [27] R. Leus. The generation of stable project plans. Ph.D. thesis, Katholieke Universiteit Leuven, Leuven, Belgium, 2003.
- [28] R. Leus and W. Herroelen. On the merits and pitfalls of critical chain scheduling. *Journal of Operations Management* 19(5):559–577, 2001.
- [29] R. Leus and W. Herroelen. Stability and resource allocation in project planning. *IIE Transactions* 36(7):667–682, 2004.
- [30] R. Leus and W. Herroelen. The complexity of machine scheduling for stability with a single disrupted job. *Operations Research Letters* 33(2):151–156, 2005.

- [31] S. V. Mehta and R. M. Uzsoy. Predictive scheduling of a job shop subject to breakdowns. *IEEE Transactions on Robotics and Automation* 14(3):365–378, 1998.
- [32] R. H. Möhring. Scheduling under uncertainty: Bounding the makespan distribution. H. Alt, ed. *Computational Discrete Mathematics: Advanced Lectures*. Springer, New York, 2001.
- [33] N. Policella, A. Oddi, and A. Cesta. Generating robust partial order schedules. *Proceedings of CP2004*. Springer, Toronto, Canada, 2004.
- [34] B. Roy. Robustesse de quoi et vis-à-vis de quoi mais aussi robustesse pourquoi en aide à la décision? J. Figueira, C. Henggeler-Anthunes, J. Climaco, eds. *Proceedings of the 56th Meeting of the European Working Group on Multiple Criteria Decision Making*, Coimbra, Portugal, 2002.
- [35] E. Sanlaville. Ordonnancement sous conditions changeantes—Habilitation à diriger des recherches. Ph.D. thesis, Université Blaise Pascal, Clermont-Ferrand, France, 2004.
- [36] F. Stork. Stochastic resource-constrained project scheduling. Ph.D. thesis, School of Mathematics and Natural Sciences, Technical University of Berlin, Berlin, Germany, 2001.
- [37] S. Van de Vonder. Proactive-reactive procedures for robust project scheduling. Ph.D. thesis, Department of Decision Sciences and Information Management (KBI), Katholieke Universiteit Leuven, Leuven, Belgium, 2006.
- [38] S. Van de Vonder, E. Demeulemeester, and W. Herroelen. Heuristic procedures for generating stable project baseline schedules. *European Journal of Operational Research*, 2007. Forthcoming.
- [39] S. Van de Vonder, E. Demeulemeester, and W. Herroelen. An investigation of efficient and effective predictive-reactive project scheduling procedures. *Journal of Scheduling*, Forthcoming.
- [40] S. Van de Vonder, F. Ballestin, E. Demeulemeester, and W. Herroelen. Heuristic procedures for reactive project scheduling. *Computers & Industrial Engineering* 52(1):11–28, 2007.
- [41] S. Van de Vonder, E. Demeulemeester, W. Herroelen, and R. Leus. The use of buffers in project management: The trade-off between stability and makespan. *International Journal of Production Economics* 97(2):227–240, 2005.
- [42] S. Van de Vonder, E. Demeulemeester, W. Herroelen, and R. Leus. The trade-off between stability and makespan in resource-constrained project scheduling. *International Journal of Production Research* 44(2):215–236, 2006.
- [43] S. Van de Vonder, E. Demeulemeester, R. Leus, and W. Herroelen. Proactive-reactive project scheduling—Trade-offs and procedures. J. Jozefowska and J. Weglarz, eds. *Perspectives in Modern Project Scheduling. International Series in Operations Research and Management Science*. Springer, New York, 2006.
- [44] M. Vanhoucke, E. Demeulemeester, and W. Herroelen. An exact procedure for the resource-constrained weighted earliness-tardiness project scheduling problem. *Annals of Operations Research* 102:179–196, 2001.
- [45] G. Vieira, J. Herrmann, and E. Lin. Rescheduling manufacturing systems: A framework of strategies, policies, and methods. *Journal of Scheduling* 6(1):39–62, 2003.
- [46] J. Wang. Constraint-based schedule repair for product development projects with time-limited constraints. *International Journal of Production Economics* 95(3):399–414, 2005.
- [47] S. D. Wu, R. H. Storer, and P. C. Chang. One-machine rescheduling heuristics with efficiency and stability as criteria. *Computers and Operations Research* 20(1):1–14, 1993.
- [48] G. Yu and X. Qi. *Disruption Management—Framework, Models and Applications*. World Scientific, Singapore, 2004.
- [49] G. Zhu, J. Bard, and G. Yu. Disruption management for resource-constrained project scheduling. *Journal of the Operational Research Society* 56(4):365–381, 2005.



# Trends in Operations Research and Management Science Education at the Introductory Level

**Frederick S. Hillier**

Department of Management Science & Engineering, Stanford University, Stanford,  
California 94305, fhillier@stanford.edu

**Mark S. Hillier**

Department of Information Systems and Operations Management, University of Washington,  
Seattle, Washington 98195, mhillier@u.washington.edu

**Abstract** The last 40 years have seen major changes in operations research and management sciences education at the introductory level. This tutorial traces the changes that have occurred, describes what subjects are currently being taught, and assesses the changes that can be expected in the near future.

**Keywords** education; educational trends; OR/MS education

---

## 1. Introduction

Although its roots go back much further, the beginning of the activity called *operations research* (or *operational research* in certain parts of the world) is generally attributed to teams of scientists who conducted research on how to perform military operations more effectively during World War II.<sup>1</sup> The activity of conducting research on how to perform activities of any kind more effectively then began to spread to business and industry during the years following World War II. Courses in operations research (OR) and then degree-granting programs in the field moved into colleges and universities during the 1950s and 1960s. Textbooks for teaching operations research at the introductory level also began to be available during the same period.

Of these initial textbooks, *Introduction to Operations Research* by Frederick S. Hillier and Gerald J. Lieberman [6] (now in its eighth edition) is still widely used today. We use the evolution of the many editions of this textbook since its initial publication in 1967 as a vehicle to describe the trends in OR education at the introductory level for students in engineering and the mathematical sciences.

Although in its early years this textbook was also widely used in business schools, specialized textbooks aimed directly at business students later were developed and used in courses that typically were entitled “Management Science” (MS) or “Quantitative Methods.” A prominent example is the Anderson-Sweeney-Williams textbook [1], *An Introduction to Management Science: Quantitative Approaches to Decision Making*, that now is in its 12th edition. Early MS textbooks such as this one covered many of the same topics as the Hillier-Lieberman textbook, including substantial coverage of algebraic modeling and basic OR algorithms, but at a mathematical level appropriate for business students. Over the past decade or so, a new generation of MS textbooks<sup>2</sup> has appeared that greatly deemphasizes this

<sup>1</sup> See Gass and Assad [4] for an informal history of operations research and its precursors dating as far back as 1564.

<sup>2</sup> The first of these new-generation textbooks, *Management Science: A Spreadsheet Approach*, by Donald Plane [8], was published in 1994.

kind of coverage and instead emphasizes spreadsheet modeling. Typical is our textbook [5], *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets*. We will use the just-published third edition of this textbook in §3 to illustrate current trends in MS education at the introductory level.

Before discussing these trends in MS education for students in business school, we first will focus on trends in introductory OR education, that is, trends in introductory OR courses for students in engineering and the mathematical sciences.

## 2. Trends in Introductory OR Education

Gass and Assad [4] identify a few important OR books that were published during the early years of the field. Notable among these were *Methods of Operations Research* by Morse and Kimball [7] and *Introduction to Operations Research* by Churchman et al. [3]. Although these

### Origins of the Hillier-Lieberman Textbook

As the first co-author of both this paper and the Hillier-Lieberman textbook, let me now present for the record a first-person account of the origins of this textbook [6].

When I arrived at Stanford University as a freshman in September, 1954, I had the great good fortune to have Jerry Lieberman assigned as my freshman advisor. Jerry took me under his wing and continued as my advisor all the way through both my undergraduate and graduate programs. During my total of seven years of study, Jerry put me into many wonderful courses with top-notch faculty. In addition to the regular undergraduate engineering courses and the full industrial engineering curriculum, I took 17 statistics courses, 13 math courses, 5 economics courses, a technical writing course, etc., as well as every operations research course given in any department. During my graduate study, Jerry also arranged for me to teach a couple of courses, including the “Introduction to Operations Research” course that had so excited me when I took it from Jerry a few years before. After auditing Harvey Wagner’s version of the same course, I threw myself into the course preparation with meticulous care and thoroughly enjoyed the experience. These class notes later were to provide the foundation for my part of the first edition of our textbook.

After receiving my Ph.D. in June of 1961, I accepted an offer to join the Stanford faculty. Jerry and I soon began discussing the need for a new introductory textbook in operations research. We decided that we wanted to develop a path-breaking textbook that would help establish the direction of education in this emerging field. I was very excited about this prospect, because of both my enthusiasm for the material and the opportunity to be a co-author with Jerry, who was already a well-known book author.

Immediately on my return from a leave at Cornell for the 1962–1963 academic year, we began working on the book. It was a labor of love that would go on for over three years, including much class testing and feedback from colleagues. The first edition then was published in 1967. The publisher was Holden-Day, a small publishing company for whom Jerry was serving as a series editor.<sup>3</sup>

Jerry and I continued to work together on the next four editions of the book published in 1974, 1980, 1986, and 1990. Tragically, in 1991, Jerry received the horrible news that he had amyotrophic lateral sclerosis (Lou Gehrig’s disease), which eventually took his life in 1999. When Jerry became ill, I promised myself that I would continue to devote myself to subsequent editions of the book. The sixth, seventh, and eighth editions were published with copyright dates of 1995, 2000, and 2005; and I am currently working on the ninth edition to be published in January, 2009. God willing, I plan to continue future editions until at least the 50th anniversary of the book in 2017, and hopefully much longer. I look forward to the challenge of continuing and enhancing the Hillier-Lieberman tradition.

<sup>3</sup> McGraw-Hill purchased the rights to our book from Holden-Day in 1988, so McGraw-Hill has been the publisher since that time.

books did not contain exercises and therefore were not fully designed for use as a textbook, the Churchman et al. book in particular did serve as a basic text for a considerable number of years. The first edition of *Introduction to Operations Research* by Hillier and Lieberman [6] followed a decade later and quickly became a standard introductory textbook. Through various editions, it maintained this status over the next 40 years to the present. The sixth edition won honorable mention for the 1995 Lanchester Prize and the eighth edition was awarded the 2004 INFORMS Expository Writing Award. The eighth edition continues to be the market leader in both the United States and internationally (various editions have been translated into well over a dozen other languages). A ninth edition is scheduled for publication in January 2009.

### 2.1. Contents of the First Edition (1967) of the Hillier-Lieberman Textbook

The first edition of the Hillier-Lieberman textbook seems somewhat quaint now, 40 years later, in light of both all the subsequent advances in the field and all the subsequent refinements in pedagogical approaches to teaching OR at the introductory level. However, it is of interest to briefly set the stage for beginning to trace the subsequent trends in OR education. (The reviewer of this tutorial commented that the first edition also is of interest because “many people argue that the 17 chapters of this edition defined OR for a generation of students.”)

Table 1 shows the table of contents of the first edition. After two introductory chapters, the next two chapters provide primers or reviews of fundamentals for dealing with probabilistic

TABLE 1. Table of contents of the first edition of the Hillier-Lieberman textbook.

Chapter	
Part I	Methodology
1	Introduction
2	Planning an Operations Research Study
Part II	Fundamentals
3	Probability Theory
4	Statistical Inference and Decision Theory
Part III	Techniques: Mathematical Programming
5	Linear Programming
6	Special Types of Linear Programming Problems
7	Network Analysis, Including PERT
8	Dynamic Programming
9	Game Theory
Part IV	Techniques: Probabilistic Models
10	Queueing Theory
11	The Application of Queueing Theory
12	Inventory Theory
13	Markov Chains and Their Applications
14	Simulation
Part V	Techniques: Advanced Topics in Mathematical Programming
15	Advanced Topics in Linear Programming
16	Integer Programming
17	Nonlinear Programming
Appendices	Convexity
	Classical Optimization Methods
	Matrices and Matrix Manipulations
	Simultaneous Linear Equations
	Tables

OR models. Chapter 4 provides an introduction to such decision analysis topics as Bayes procedure and posterior probabilities but most of the chapter is devoted to traditional topics of statistical inference.

Chapters 5 to 9 give an introduction to some basic areas of mathematical programming. Chapter 5 presents the linear programming model and its underlying assumptions, describes the simplex method (including the Big M method and a brief mention of the two-phase method), and briefly introduces duality theory and sensitivity analysis. Chapter 6 covers the transportation problem, the transshipment problem, and the assignment problem, including specialized algorithms for solving these problems. Chapter 7 presents the maximum flow and shortest path problems (but not the general minimum cost flow problem), as well as the minimum spanning tree problem and PERT. Chapter 8 describes the general characteristics of dynamic programming problems and presents various examples of both deterministic and probabilistic dynamic programming. Chapter 9 focuses almost exclusively on two-person, zero-sum games.

Chapters 10 to 14 then turn to an introduction to basic types of probabilistic models. Chapter 10 focuses mainly on Markovian queueing models, including even priority models, but only briefly mentions queueing networks. Chapter 11 then deals with various aspects of applying queueing theory, including cost models for designing queueing systems. Chapter 12 presents a variety of classic inventory models, including both deterministic and stochastic models. Chapter 13 is devoted mostly to discrete-time Markov chains, but also includes a relatively brief discussion of both Markov decision models and continuous-time Markov chains. Chapter 14 introduces the various techniques of simulation, including the generation of random numbers, the generation of random observations from a probability distribution, and variance-reducing techniques.

Chapters 15 to 17 turn to some “advanced topics” in mathematical programming, so the implication is that these topics need not be included in a basic introductory survey course. Chapter 15 expands substantially on the treatment of duality theory and post-optimality analysis in Chapter 5. It also covers such topics as the revised simplex method, the dual simplex method, the decomposition principle, stochastic programming, and chance-constrained programming.

What is perhaps most striking about the table of contents in Table 1 is that such basic topics today as integer programming and nonlinear programming are relegated to the “advanced topics” at the end of the book. However, back in the mid-1960s, these were indeed fairly new and relatively advanced topics. Chapter 16 features Gomory’s cutting plane algorithm but also introduces the branch-and-bound approach to integer programming that was quite new at the time. Chapter 17 presents the Kuhn-Tucker conditions (now known as the Karush-Kuhn-Tucker or KKT conditions), applies these conditions to quadratic programming with Wolfe’s modified simplex method, and discusses separable programming. It then briefly surveys algorithmic approaches to convex programming, including Fiacco and McCormick’s then-new sequential unconstrained minimization technique.

The pedagogical style of the first edition is a fairly formal one. Many chapters or sections begin with a mathematical statement of the general model being considered and then attention is turned to an algorithm for solving the model. One or more numerical examples are used to illustrate the formulation of the model and the application of the algorithm. This relatively concise treatment of the material results in a trim 6” × 9” book of 639 pages.

## **2.2. The Evolution of Subsequent Editions of This Textbook**

Subsequent editions underwent numerous revisions, both large and small, to reflect developments in the field and changing pedagogical tastes. We will briefly summarize the most important changes.

Starting with the second edition, the pedagogical style of the book was changed fairly substantially from the one of the first edition. Much more emphasis was given to developing

interesting and relatively realistic examples. Rather than beginning many chapters or sections with a general mathematical statement of the model being considered, a fairly elaborate prototype example was used instead to introduce the topic and then to illustrate the formulation of the model and the execution of an algorithm. The more expansive treatment of the material and the addition of many more problems (and then cases) considerably increased the size of the book.

Many new topics also were added to new editions of the book. Some of these involve well developed, but fairly specialized, areas of operations research, including goal programming, the minimum cost flow problem, the network simplex method, forecasting, reliability theory, and the theoretical foundations of the simplex method. Other topics were added to reflect important new developments in the field. These include the interior-point approach to solving linear programming problems, the branch-and-cut approach to solving integer programming problems, constraint programming, metaheuristics, and multiechelon inventory models for supply chain management. In §2.3, we will discuss the importance of these five new topics and why instructors should consider at least briefly including them in introductory OR courses.

Some topics that were covered only briefly in the first edition were greatly expanded and revised in subsequent editions. These include such key topics as decision analysis, duality theory and sensitivity analysis, simulation, Markov decision processes, and project management with PERT/CPM.

In addition, the rather short chapters on integer programming and nonlinear programming that were relegated to the back of the book as “advanced topics” in the first edition subsequently were greatly expanded and brought forward as mainstream chapters. After deleting the section on Gomory’s cutting plane algorithm, the integer programming chapter now includes a section describing some real-world applications, two sections on model formulation, and extensive coverage of the branch-and-bound approach, as well as sections on the branch-and-cut approach and the incorporation of constraint programming. Similarly, the nonlinear programming chapter was expanded into a relatively comprehensive survey of the area.

Table 2 shows the timeline for when various major new topics were added to a new edition for the first time. (In a few cases, these topics were subsequently moved to the supplements on the CD-ROM in the most recent editions to save space.) Thus, the table gives an indication of the trends in OR education at the introductory level over the last few decades.

All these changes greatly increased the size of the book. Up to the seventh edition, each new edition was significantly larger than the preceding one. In contrast to the first edition that has 639 pages, the seventh edition had grown to 1,214 pages with trim size expanded from 6” × 9” to 8” × 9”. Although the expanded material provides considerable flexibility to instructors for what to cover, a somewhat smaller size might be preferable for a textbook for introductory survey courses. Therefore, a strong effort was made to reduce the size of the book. This succeeded in reducing the eighth edition to 1,061 pages (with an 8” × 10” trim size), despite adding nearly 100 pages of new material on recent developments in the field. This reduction was accomplished largely by transferring various little-used sections and chapters in preceding editions (as well as many cases) to the book’s CD-ROM and website as supplements.

The fifth edition introduced another key change by packaging software with the book for the first time. For the first four editions, the students had been expected to use paper and pencil to do their homework for learning how to execute the various algorithms. Mark S. Hillier then developed a tutorial software package to accompany the fifth and sixth editions. This package featured demonstration examples, interactive routines, and automatic routines for the various algorithms in the book. Each interactive routine enables the student to execute the corresponding algorithm interactively, making the needed decision at each

TABLE 2. Major new topics in new editions of the Hillier-Lieberman textbook.

Edition	Major new topics
Second (1974)	Tabular form of the simplex method A “fundamental insight” for the simplex method Multidivisional and multitime period problems Forecasting A complete chapter on decision analysis A complete chapter on Markov decision processes Reliability Regenerative method of statistical analysis for simulation Branch-and-bound algorithms for binary and mixed integer programming Gradient search procedure for unconstrained optimization
Third (1980)	Goal programming Special formulation techniques for linear programming Special formulation techniques for integer programming One-dimensional procedure for unconstrained optimization
Fourth (1986)	A complete chapter on reliability Greatly expanded chapter on nonlinear programming A continuous-review stochastic inventory model
Fifth (1990)	Interior-point approach to solving linear programming problems Use of microcomputer software Minimum cost flow problem Network simplex method Branch-and-cut approach to solving integer programming problems Jackson queueing networks Forecasting with seasonal effects
Sixth (1995)	Case studies of real applications Several case problems Utility theory
Seventh (2000)	Incorporation of MPL, CPLEX, LINDO, LINGO, Excel spreadsheets, etc. Eight new sections on the practice of operations research Many case problems A complete chapter on project management with PERT/CPM
Eighth (2005)	Metaheuristics (a complete new chapter) Constraint programming Multiechelon inventory models for supply chain management Spreadsheet modeling Hungarian algorithm for the assignment problem Newton’s method for unconstrained optimization Many supplementary “worked examples” on the CD-ROM A test bank for instructors

step while the computer does the needed arithmetic. The result is a far more efficient and effective learning process that also is more stimulating to the student.

For the seventh and subsequent editions, a professional software company (Accelet Corp.) further enhanced this package of interactive and automatic routines and implemented it in Java 2, so it is platform independent. Michael O’Sullivan implemented the package of demonstration examples in JavaScript. Starting with the seventh edition, a wealth of other professional software packages were bundled with the book together with numerous illustrations of their use. The packages accompanying the eighth edition (and anticipated for the ninth edition) include the following:

- Interactive Operations Research (IOR) Tutorial (the package of interactive and automatic routines described above).

- OR Tutor (the package of demonstration examples of various algorithms mentioned above).
- Several Excel add-ins, including Premium Solver for Education (an enhancement of the basic Excel Solver), TreePlan (for decision analysis), SensIt (for probabilistic sensitivity analysis), RiskSim (for simulation), and Solver Table (for automating sensitivity analysis in optimization problems).
- Student versions of MPL (an algebraic modeling language) and three of its solvers: CPLEX (the most widely used state-of-the-art optimizer), CONOPT (for convex programming), and LGO (for both global optimization and convex programming).
- Student versions of LINDO (a traditional optimizer) and LINGO (an algebraic modeling language).
- Queueing Simulator (for the simulation of queueing systems).
- Crystal Ball Professional Edition (for risk analysis, including especially simulation).

### 2.3. Some Key New Topics for Introductory OR Courses

Over the 40 years since the publication of the first edition, many important new developments have occurred in the field. We highlight five here that we feel deserve strong consideration for inclusion at least briefly in even an introductory OR survey course. Thus, all five are covered in some detail in the most recent edition(s) of the Hillier-Lieberman textbook.

- **Interior-Point Approach to Linear Programming**

The most significant new development during the 1980s was the discovery of the efficiency of the interior-point approach to solving linear programming problems. Much more progress has been made in more recent years in developing algorithms of this type (commonly referred to as *barrier algorithms*). Although this approach did not supplant the simplex method (or dual simplex method) as the method of choice for problems of moderate size, it often is the most efficient method (and perhaps the only tractable method) for solving huge problems. Therefore, although its technical details are quite complicated, it is appropriate to introduce introductory students to its main concepts.

- **Branch-and-Cut Algorithms for Integer Programming**

For many years, *branch-and-bound algorithms* provided the most efficient way of solving integer programming problems. However, in more recent years, this kind of algorithm has been supplanted by branch-and-cut algorithms that incorporate the branch-and-bound approach into a much broader framework that also includes automatic problem preprocessing and the generation of cutting planes. Students should be made aware that this kind of algorithm now provides the state-of-the-art approach to solving integer programs.

- **Constraint Programming Techniques for Mathematical Programming**

An exciting recent development is the discovery of constraint programming techniques that are beginning to greatly expand our ability to formulate and solve certain kinds of mathematical programming problems, including integer programming and combinatorial optimization problems. Although the further development of this approach is still an active area of research, it is time to begin briefly introducing this key new concept in introductory courses.

- **Metaheuristics for Complex Problems**

Despite the continuing progress in developing more efficient algorithms, the problems that arise in practice often are too large and complex to be solved to optimality. Therefore, *heuristic methods* are frequently used to search for a very good, but not necessarily optimal, solution. This has led to the development of several powerful metaheuristics (e.g., tabu search, simulated annealing, and genetic algorithms), which are general solution methods that provide both a general structure and strategy guidelines for developing a specific heuristic method to fit a particular kind of problem. Because of their wide usage in practice, it is important for students to be made aware of the existence of these metaheuristics and

perhaps to be introduced to their general concepts. Therefore, the eighth edition of the Hillier-Lieberman textbook added a complete new chapter on metaheuristics.

#### • Multiechelon Inventory Models

In our growing global economy, effective *supply chain management* is now a key success factor for many leading companies. Therefore, multiechelon inventory models are a vital tool for managing the inventories at the stages of a company's supply chain in a coordinated way. Consequently, any survey of inventory models in an introductory course now should at least introduce multiechelon models.

In addition to these five key topics, there are other recent developments that also warrant some consideration for inclusion in an introductory OR survey course. At the time of this writing, several of these are being considered for addition to the ninth edition.

### 3. Trends in Introductory MS Education

Section 2 focused on trends in introductory OR education. We now turn to discussing the trends in business schools, where the corresponding courses commonly replace the name "operations research" (OR) by "management science" (MS). After briefly describing the history of introductory MS education, we will use our own [5] textbook (the third edition of *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets*) to illustrate the current trends, including current progress in spreadsheet modeling.

#### 3.1. A Brief History of Introductory MS Education

Management science topics were introduced into the curriculum of many business schools as early as the 1960s (and even earlier in some cases). The Hillier-Lieberman OR textbook (among others) was widely used in business schools during the early years after its initial publication in 1967. However, it then began to be replaced during the 1970s as textbooks directly aimed at business students (such as the classic Anderson-Sweeney-Williams [1] textbook) started to appear. These MS textbooks covered most of the same topics as the Hillier-Lieberman OR textbook (including coverage of basic algorithms), but at a considerably lower mathematical level and with more emphasis on examples of business applications.

In one sense, the 1980s were the glory days for introductory MS education in the United States. Because the Association to Advance Collegiate Schools of Business (AACSB) required the inclusion of a management science course in the core MBA curriculum to obtain accreditation, huge numbers of business students were taking a management science course (sometimes labeled "quantitative methods") and a considerable number of management science faculty were being hired to teach these courses. However, it seemed that an increasing number of students were struggling unsuccessfully to penetrate the "algebraic curtain" that was raised in many of these courses when presenting the algebraic models and algorithms of operations research, and there was considerable skepticism among both the students and other faculty about the relevance of these courses. Perhaps one factor, according to a common perception of instructors and the periodic results of international testing of the mathematical ability of students in various countries, was that the mathematical background and ability of typical American business students was becoming progressively weaker over time.

A catastrophic event occurred in 1991 when the AACSB dropped management science from its core body of knowledge needed to receive accreditation. Over the next several years, most business schools dropped the requirement that its students must take a management science course. A smattering of management science concepts might appear in other courses (such as a required operations management course) or even as one portion of a core course, but that was all. A management science course usually was still available as an elective, but most students were not taking it at most schools. The size of many management science faculties began to shrink.



As a result, the management science community was forced to take a hard look at what went wrong with old ways of teaching management science and what changes needed to be made. In 1996, INFORMS Business School Education Task Force issued a report that included the following statement.

There is clear evidence that there must be major change in the character of the (introductory management science) course in this environment. There is little patience with courses centered on algorithms. Instead, the demand is for courses that focus on business situations, include salient nonmathematical issues, use spreadsheets, and involve model formulation and assessment more than model structuring. Such a course requires new teaching materials. ([9], p. 40)

This statement implied several key messages:

1. “There is little patience with courses centered on algorithms” suggests that it was a mistake for traditional courses and textbooks to focus largely on algorithms because they are not relevant for future managers (typical business students).
2. Teaching algorithms only succeeds in turning off these students and diverting attention from the relevant concepts of management science.
3. “The demand is for courses that focus on business situations (and) include salient nonmathematical issues” suggests that a **case studies approach** would be an appropriate way to consider the business and nonmathematical issues when applying management science.
4. “The demand is for courses that... involve model formulation and assessment more than model structuring” suggests that a **modeling approach** is appropriate, but it should be one that focuses on modeling on a conceptual and evaluation level rather than on the algebraic details of the model.
5. “The demand is for courses that... use spreadsheets” seems to suggest that the emphasis should be on **spreadsheet modeling** instead of algebraic modeling.

The statement ends with the plea that the kind of course being recommended “requires new teaching materials.” This plea was answered in the mid-to-late 1990s with the publication of several new introductory management science textbooks that emphasize spreadsheet modeling. In most cases, the approach is to present a large number of relatively brief examples of the application of the mechanics of spreadsheet modeling to a variety of business problems, but not to present case studies (except for end-of-chapter cases) that illustrate the role of this process in an overall managerial study. However, other books also were being published that provide this broader perspective from the viewpoint of a corporate general manager. One notable example is the book by Peter Bell [2] entitled *Management Science/Operations Research: A Strategic Perspective*.

We believe that our own textbook (first published in 1999 and now in its third edition), provides a unique, thoroughly modern approach. As implied by its title, *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets*, we feature the use of spreadsheet modeling within the context of case studies. We draw on our own experience with this book when describing the current trends in introductory MS education in the next subsection.

Before concluding this brief history of introductory MS education, we should point out that the AACSB again revised its accreditation standards in 2003 to require the coverage of “statistical data analysis and management science” in MBA courses. However, the standard does not require that a full course be devoted to management science, or even to both statistical data analysis and management science. It is still unclear in 2007 how much impact this revised guideline will have on the resurgence of management science in business school curricula.

### 3.2. Current Trends in Introductory MS Education

In this section we elaborate further on the current trends in introductory MS education and their rationale as illustrated by our own textbook.

- **Use Spreadsheets**

The modern approach to teaching of management science clearly is to use spreadsheets as a primary medium of instruction. Inasmuch as both business students and managers now live with spreadsheets, they provide a comfortable and enjoyable learning environment. Modern spreadsheet software, including Microsoft Excel, can now be used to do real management science. In addition to Excel's Solver and Frontline System's premium versions of Solver for doing optimization within spreadsheets, add-ins to Excel such as TreePlan or Precision Tree for decision trees and Crystal Ball, @RISK, or Risk Solver for simulation allow performing decision analysis and simulation modeling efficiently within the spreadsheet environment. For student-scale models (which include many practical real-world models), spreadsheets are a much better way of implementing management science models than traditional algebraic solvers. The algebraic curtain that was so prevalent in traditional management science courses and textbooks now can be lifted.

However, with the new enthusiasm for spreadsheets, there is a danger of going overboard. Spreadsheets are not the only useful tool for performing management science analyses. Occasional modest use of algebraic and graphical analyses still have their place and it would be a disservice to not develop the skills of students in these areas when appropriate. Furthermore, a management science textbook should not be mainly a spreadsheet cookbook that focuses largely on spreadsheet mechanics. Spreadsheets are a means to an end, not an end in themselves.

- **Use a Modeling Approach**

Model formulation lies at the heart of management science methodology. Therefore, it is appropriate to emphasize heavily the art of model formulation, the role of a model, and the analysis of model results. The modern approach is to do this by primarily (but not exclusively) using a spreadsheet format rather than algebra for formulating and presenting a model.

Some instructors have many years of experience in teaching modeling in terms of formulating algebraic models (or what the INFORMS Task Force called "model structuring" in the statement quoted in §3.1). Some of these instructors feel that students should do their modeling in this way and then transfer the model to a spreadsheet simply to use the Excel Solver to solve the model. However, most business students find it more natural and comfortable to do their modeling directly in a spreadsheet. Furthermore, by using the best spreadsheet modeling techniques, formulating a spreadsheet model tends to be considerably more efficient and transparent than formulating an algebraic model.

Another break from tradition is to virtually ignore the algorithms that are used to solve the models. There is no good reason why typical business students should learn the details of algorithms executed by computers. Within the time constraints of a one-term management science course, far more important lessons are to be learned. High on this list is the art of modeling managerial problems on a spreadsheet.

Formulating a spreadsheet model of a real problem typically involves much more than designing the spreadsheet and entering the data. Therefore, it is important to work through the process step by step: understand the unstructured problem, verbally develop a structure for the problem, gather the data, express the relationships in quantitative terms, and then lay out the spreadsheet model. The structured approach highlights the typical components of the model (the data, the decisions to be made, the constraints, and the measure of performance) and the different types of spreadsheet cells used for each. Consequently, the emphasis should be on the modeling rather than spreadsheet mechanics.

- **Use a Case Studies Approach**

However, an emphasis on spreadsheet modeling would be quite sterile if a course involved little more than examining a long series of brief examples with their spreadsheet formulations. In addition to examples, it is helpful to include case studies patterned after actual applications to convey the whole process of applying management science. By drawing the student into the story, a case study can bring the application of management science to life in a context that illustrates its relevance for aiding managerial decision making. This case-centered approach should make the material more enjoyable and stimulating while also conveying the practical considerations that are key factors in applying management science. Case studies are a key to preparing students for the practical application of management science in all its aspects.

### **3.3. Some Key New Topics for Introductory MS Spreadsheet Modeling Courses**

With the increasing emphasis on spreadsheet modeling in modern introductory MS courses, it now is useful to introduce a few new topics into these courses that take advantage of the full power of modern spreadsheet technology. We describe some key topics of this kind below.

- **Spreadsheet Engineering**

As spreadsheet modeling has matured and become the dominant medium of instruction in MS education, there has been an increased emphasis on the design of spreadsheets, commonly referred to as spreadsheet engineering. Although one of the greatest benefits of spreadsheets is their flexibility, this benefit also leads to one of their greatest dangers as a modeling tool. It is extremely easy to create unorganized spreadsheet models that are difficult to interpret, difficult to understand, and difficult to debug. This unorganized approach often leads to spreadsheets that contain critical errors. An entire chapter of our textbook is devoted to the art of modeling with spreadsheets. The chapter discusses the process of modeling and provides tips on creating good spreadsheet models that are well organized, easy to interpret, and simple to debug. The tips include (1) how to organize or sketch out a spreadsheet, (2) start small before expanding to full size, (3) separate the data from the formulas, (4) use borders or shading to distinguish data cells, changing cells (decision variables), and the target cell (objective function), and (5) use range names to make formulas easier to read.

- **Automated Sensitivity Analysis**

Traditionally, teaching sensitivity analysis in linear programming has emphasized the interpretation of shadow prices for the constraints and the allowable ranges for changes in the objective coefficients as determined by using a sensitivity report. Although this information is still important and useful, one of the great assets of spreadsheets is how easy it is to make changes to a model. The new solution is then just one click of the Solve button away. Moreover, many current textbooks come with add-ins to Excel such as Solver Table in our book, which can automate this type of sensitivity analysis. The add-ins make it easy to build a table that shows how the solution changes over a range of possible values for one or two of the parameters of the problem. Solver Table also works with integer or nonlinear programming problems and can analyze the impact of changes in *any* parameter of a model, not just objective coefficients or constraint right-hand sides.

- **Search Procedures**

Frontline Systems, the original developer of the standard Solver that Microsoft includes with Excel, developed premium versions of Solver. They graciously made the Premium Solver for Education available with many textbooks, including our own. This solver includes a new search procedure called Evolutionary Solver that uses a genetic algorithm to solve difficult nonlinear problems (e.g., nonsmooth and/or discontinuous problems). Because this

new search procedure uses the same simple Solver interface, it is easy to incorporate the use of these advanced algorithms into even introductory MS courses.

#### • Simulation with Spreadsheets

Much like the incorporation of optimization algorithms like Solver into spreadsheets has revolutionized the way optimization is taught to business students, software packages like Crystal Ball, @RISK, and Risk Solver that bring Monte Carlo simulation into the spreadsheet environment have revolutionized the teaching of simulation to business students. It is easy to take any financial spreadsheet and add the power of simulation to analyze the effect of uncertainty. An exciting new topic in MS education is doing optimization with simulation. For example, the OptQuest module in Crystal Ball uses a combination of scatter search, tabu search, and neural networks to optimize a set of decision variables where the objective function is based on simulation outcomes.

## 4. Conclusions

Over the last few decades, there has been a steady evolution in the content of introductory OR and MS courses. During the early years, heavy emphasis was given to the mathematical models and algorithms of operations research. This has continued to be the emphasis in courses directed to students in engineering and the mathematical sciences, but with some updating to include such new developments as the interior-point approach to linear programming, branch-and-cut algorithms for integer programming, constraint programming techniques for mathematical programming, metaheuristics for complex problems, and multi-echelon inventory models. However, there has been a growing realization over the years that this emphasis on mathematical models and algorithms is not the appropriate one for typical business school students who will become managers rather than applied mathematicians. Since the mid-1990s, the new wave in the teaching of management science is to emphasize spreadsheet modeling and perhaps case studies instead.

## References

- [1] D. R. Anderson, D. J. Sweeney, and T. A. Williams. *An Introduction to Management Science: Quantitative Approaches to Decision Making*, 12th ed. Thomson/South-Western, Mason, OH, 2008.
- [2] P. C. Bell. *Management Science/Operations Research: A Strategic Perspective*. South Western College Publishing, Mason, OH, 1999.
- [3] C. W. Churchman, R. L. Ackoff, and E. L. Arnoff. *Introduction to Operations Research*. Wiley, New York, 1957.
- [4] S. I. Gass and A. A. Assad. *An Annotated Timeline of Operations Research: An Informal History*. Kluwer Academic Publishers, Boston, MA, 2005.
- [5] F. S. Hillier and M. S. Hillier. *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets*, 3rd ed. McGraw-Hill/Irwin, Burr Ridge, IL, 2008.
- [6] F. S. Hillier and G. J. Lieberman. *Introduction to Operations Research*, 8th ed. McGraw-Hill, Boston, MA, 2005.
- [7] P. M. Morse and G. E. Kimball. *Methods of Operations Research*. MIT Press and Wiley, New York, 1951.
- [8] D. R. Plane. *Management Science: A Spreadsheet Approach*. Boyd & Fraser, Danvers, MA, 1994.
- [9] Report of the Operating Subcommittee of the INFORMS Business School Education Task Force. *OR/MS Today* (February):36–41, 1997.

# Business Engineering: A Practical Approach to Valuing High-Risk, High-Return Projects Using Real Options

*Scott Mathews and Jim Salmon*

The Boeing Company, Seattle, Washington 98124 {scott.h.mathews@boeing.com,  
jim.salmon@boeing.com}

**Abstract** Technologists and engineers endeavor to design and propose leading edge concepts, but the development of these concepts ultimately depends on obtaining funds justified by a business case. Most existing business case tools and methods, which have their origins in the conservative banking industry, tend to favor those project concepts that have secure annuity-like returns such as extensions of existing product lines. This tutorial provides engineers with the business case methods and tools to calculate the value of smaller, more risky projects where new technology or markets are involved, and which potentially offer higher returns in the long run. These straightforward methods and tools have been adopted from sophisticated techniques used in the options markets, where investments in risky securities are routinely traded. We first present an example scenario for a new product and review a typical business case using net present value analysis. Next we develop a “what-if” multi-scenario business case model using Monte Carlo simulation. Then we examine an investment decision using a decision tree to capture scenario flexibility. Finally, we determine a risk-averse investment decision using real options calculated with an intuitive and transparent algorithmic tool. In conclusion we show business engineering as a new approach that provides engineers with investment and risk modeling tools and methods that can be incorporated alongside standard systems engineering design modeling techniques to justify the targeting of project investment dollars to manage risk, shape value outcomes, and make better strategic decisions.

**Keywords** real options; investments; risk; Black-Scholes; Datar-Mathews method; business engineering; strategic decisions; Monte Carlo simulation; systems engineering; spreadsheet modeling; valuation; capital budgeting; NPV finance

---

Too often technologists and engineers endeavor mightily to design and propose leading-edge concepts, only to have them left unfunded and on the shelf as a result of a seemingly injudicious business-case analysis. *Business engineering*<sup>1</sup> is a new approach to provide engineers with much needed financial-investment and risk-modeling tools and methods that can be incorporated alongside standard engineering design-modeling techniques. Business engineering extends engineering, particularly systems engineering risk management, practices to include additional design constraints variables like financial risk and value, and injects new tools such as targeting investment dollars to manage risk, shape value outcomes, and make better strategic decisions. Finally, business engineering provides the vocabulary for engineers to justify investments in leading-edge high-risk, high-return projects.

To provide perspective, it is instructive to understand that financial-analysis techniques have historically undergone cycles of change, just as in engineering fields where new technologies supplant older ones. The earliest approach, called *payback*, simply counted the

---

<sup>1</sup> Business engineering derives its name from *financial engineering*, where mathematicians and engineers apply their skills to pricing of risky investment contracts in the financial options markets (<http://en.wikipedia.org/wiki/computational.finance>). Business engineering uses many of the same techniques and skills, but for valuing of investments in risky, but high-return projects within corporations.

number of years of estimated profit required to return the original investment. Starting in the middle of last century following on its successful application within the banking industry, net present value (NPV) was applied to project valuation (sometimes called *capital budgeting*) and became the corporate finance standard, applying discounting techniques for future profit cashflows. NPV remains the standard in most corporate business-case analyses, even though appropriate application within the secure environment of the banking industry imperfectly transfers to riskier corporate project analysis. However, since the mid-1980s, the rise of the importance of the options in the capital markets, and, in particular the creation and wide-spread use of the Black-Scholes formula,<sup>2</sup> has led to a revolution in the way financial analysis is conducted. Option-valuation techniques are well suited to evaluating investments with flexibility, critical decision points, and major discontinuities such as one finds in high-risk, high-return technology projects. The practice of option techniques applied to business-investment decisions is termed *real options* because it applies to real product and technology assets, rather than capital market financial assets.

Real options has, however, been slow to develop because of the complexity of the techniques that have been borrowed from the capital markets and the resultant mismatch to the needs and realities of corporate financial analysis and strategic decisions. Such complexity and the resulting challenge of getting senior-management acceptance, has been a major barrier to wider corporate adoption of real-option techniques.

However, recently Boeing has developed a real-option method of valuation, referred to as the Datar-Mathews (DM) method [3]. Although algebraically equivalent to the Black-Scholes formula, it uses information that arises naturally in a standard project financial valuation, see Datar and Mathews [4] and Mathews et al. [5].

## 1. An Investment Decision: The NPV Most Likely Business Case

In order to illustrate a real-option valuation, let us first examine a project using a simple business case analyzed from an NPV approach. Imagine Boeing has the opportunity to design and build a small aircraft, somewhat smaller than the size of a 737, specialized for air-freight transportation. There is a rapidly expanding market for high-value goods to be quickly shipped from local airports near a manufacturer directly to one close to consumer-market outlet locations and vice versa. Historically, air-freight planes have been created by converting older passenger planes. Inefficiency in design and weight of the converted air freighters result in the cost of transported cargo, as measured by dollars per ton-mile, not being competitive with freight transported by overland truck. Although a new specialized air freighter might be competitive for transporting luxury goods, it remains a risky proposition because it would compete directly with the inexpensive, albeit inefficient, converted freighters. Furthermore, the air freighter market is difficult to forecast for many reasons; e.g., it is sensitive to business cycles and volatile fuel costs.

The actual business case for the air freighter is complex, involving many factors. However, the concepts presented in this paper can be sufficiently illustrated with a simplified business case. Table 1 sets forward example projections of revenues and costs using a most likely scenario. The engineers and marketing analysts are requesting authorization to spend \$100 M over the next three years. The engineers will focus on a preliminary design, particularly on nonrecurring cost-manufacturing efficiencies, to reduce the freighter's costs to make it as competitive as possible, whereas marketing needs to determine the size and price elasticity of the air-freighter market place. After three years, contingent on the success of the engineering efforts and a promising market forecast, Boeing would have to spend an estimated \$2.0 B to launch the air freighter. These one-time nonrecurring launch costs are intended for final

<sup>2</sup> For a lighter review of the Black-Scholes formula, see <http://www.risklatte.com/features/quantKnow050905.php>.

TABLE 1. NPV most likely business case.

	Year										
(\$M)	0	1	2	3	4	5	6	7	8	9	10
Target unit price (\$)											
Target unit (240) cost (\$)	20										
Unit cost (\$)					45	33	27	24	21	20	19
Unit quantity					15	30	45	60	60	60	60
Revenues (\$)					525	1,050	1,575	2,100	2,100	2,100	2,100
Recurring costs (\$)					676	977	1,212	1,412	1,284	1,200	1,139
Most likely op profits (\$)		0	0	0	(151)	73	363	688	816	900	961
Launch cost (\$)		0	0	(2,000)							
R&D expenses (\$)	(100)										

TABLE 2. NPV most likely project valuation.

Discount rate assumptions	
Project risk rate	15.0%
NPV calculations	(\$M)
PV <sub>0</sub> operating profits	\$1,126
PV <sub>0</sub> launch costs	(\$1,315)
R&D expenses	(\$100)
Total project NPV value	(\$289)

design, manufacturing readiness, Federal Aviation Administration certification and marketing outlays. The estimated unit sales, price, and unit-recurring cost depend on assumptions about product strategy and market reception.

Calculation of the NPV value is straightforwardly based on the most likely value estimates for the business-case variables. Net profits are the difference between the operating profits (sales revenue minus unit-recurring cost {[unit price – unit cost] \* unit quantity}) and the nonrecurring launch cost. Using the most likely set of estimated values, and applying the Excel NPV function<sup>3</sup> using the project discount rate of 15% (representing the rate established by management to meet a required rate of return for project investments), the project net present value at Year 0, today, is estimated to be a negative \$289 M; see Table 2. Under standard NPV decision making, the finance department’s forecast for this project is that it will lose money, and therefore the request for initial funding of \$100 M would be denied. Following that guidance, senior management would terminate the air-freighter project and appropriately direct engineering and marketing R&D funding resources to other projects with positive NPV values.

Given the uncertainty of the market and the cost estimate forecast, and thus of the project outcome, we may doubt the conclusion of the NPV analysis. An NPV analysis reduces all available information to a single scenario, eliminating other less probable, although still plausible scenarios. The *what-if scenario analysis*, a commonly used engineering technique, could elicit several plausible, but lower-probability, outcomes.

Because of the singular scenario (or path) focus, the NPV approach also has a tendency to impact project planning by channeling engineering resources early into executing a single track effort well before the resolution of significant project uncertainties. This can lead

<sup>3</sup> According to a financial economic insight, the spreadsheet-based NPV function has the effect of simultaneously summing the cash flow values while adjusting the out-year values to equivalent time- and risk-adjusted values of the target year.

to situations where in spite of good intentions, the project is directed down a course of action that limits its ability to respond with agility to changes in the engineering and technology awareness or market factors that have a potential material impact on the outcome of the project. More realistic project planning incorporates contingency plans under a multi-scenario planning approach in the event of unexpected technology or market developments. Capturing the value of project flexibility in the face of uncertainty requires a different valuation approach.

## 2. What-If Multiscenario-Modeling Approach Using Monte Carlo Simulation

Monte Carlo simulation can be advantageously applied to making investment and risk decisions for business cases. Just as this technology extends the ability to investigate what-if scenarios on matters of engineering concern, such as performance and operating stability, this same technology can be applied to what-if scenarios for price, unit sales, and cost for the air-freighter business case. In fact, Boeing applies the term “business engineering” to the more advanced models, which combine both the business and engineering aspects of a project. These advanced models effectively automate the what-if scenario analysis by application of Monte Carlo simulation. What is required, is a valuation approach that effectively uses this technology and provides a more useful estimate for the project.

Monte Carlo simulation offers the ability to incorporate into the analysis several scenarios, including those that are plausible albeit lower probability, but potentially consequential to the outcome of the project. A substantial portion of the information about these other scenarios may already be available within the corporation. For example, good engineering practice often includes calculating best- and worst-case performance outcomes for technology and products, which can be incorporated into a business-engineering analysis (see Appendix IV). Additionally, a multiscenario approach includes flexibility and critical decision points for managing the project. The underlying reality is that as events unfold prior to the launch date and one or another technology or market opportunity scenario begins to play out, decision managers have the ability to increase project value by identifying and responding to the changes.

The multiple-scenario deliberations need to focus on the high-level risks and opportunities, those that impact 10% or more of the total value of the project. There are several reasons for this. One is the obvious need to reduce the complexity and quantity of the scenarios. This is in line with standard engineering practices for a first- and second-draft design effort, where the focus is on those components that contribute 10% or more of the targeted function. Furthermore, many of the minor risks can be managed by a good engineering and management team when they become apparent.<sup>4</sup> Finally, the future is not divivable, and the immediate circumstances will predictably change—the project manager simply needs to be sufficiently flexible to respond to unfolding events to take advantage of the opportunities that might arise.

Begin the multiscenario modeling process by envisioning three scenarios: optimistic, most likely, and pessimistic. The most likely scenario is usually the scenario already derived by the finance analysts for the NPV business case. This scenario forecasts the most likely cost and revenue cash flows that would materialize if events play out as a majority of experts expect given both engineering efforts and marketing response.

The pessimistic and optimistic scenarios are derived from deliberations with a similar set of experts (technology, engineering, marketing, finance, and management) but now attempt to reflect key insights to the major project contingencies, the high-level risks and opportunities that potentially impact 10% or more of the value of the project; see Table 3.

<sup>4</sup> Note this thinking is analogous to how insurance is employed to cover large risks, whereas we accept payment responsibility for minor unpredictable incidents whose costs fall under the insurance deductible.

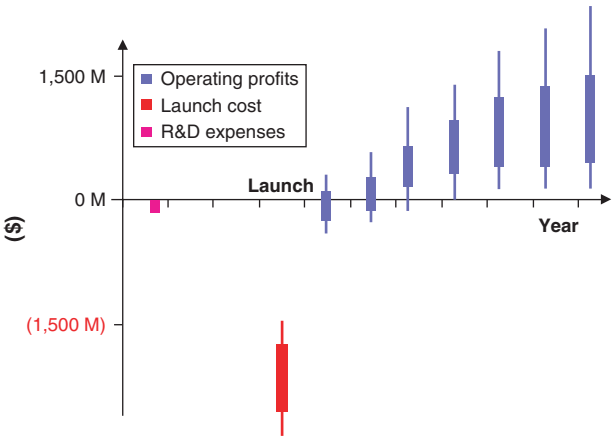


TABLE 3. Variables for various scenarios.

Contributor	Variable (\$M)	Pessimistic	Most likely	Optimistic
Engineering	Unit (240) cost	\$23	\$20	\$18
Engineering	Launch cost	\$2,500	\$2,000	\$1,500
Engineering	Production ramp	15/Year	15/Year	15–30/Year
Marketing	Unit price	\$30	\$35	\$40
Marketing	Unit quantity	270	330	555
Finance	Discount rate	15% Project; 5% Investment		
Management	Outlook	10%	Most likely	10%
Technology	R&D costs	\$100	\$100	\$100

The pessimistic scenario is determined by estimating cost and revenue cash flows that nine out of ten experts believe will be exceeded; in other words, only one of the experts (10%) believes that the pessimistic scenario will materialize. The optimistic scenario is similarly determined. Only one out of ten experts believe the estimated optimistic cost and revenue cash flows will materialize, whereas the remaining nine believe the actual cash flows will be less; see Figure 1. When ten experts are not available, assemble fewer, e.g., five, but again be sure to survey for the outlying estimates to capture a broad range of possible outcomes. This scenario and probability elicitation method is analogous to a Delphi method for systematic interactive forecasting approach based on inputs from experts. A cat-whisker bar graph illustrates the variability of the cash flows of the optimal, most likely, and pessimistic ranges for each year (with the thicker middle section running from approximately the 20th to the 80th percentiles of the distribution); see Figure 1.

FIGURE 1. Business-case scenario cash flow variability.



For each of the various scenarios, the marketing specialists can help quantify unit-quantity and unit-price forecasts.<sup>5</sup> Each scenario, i.e., optimistic, most likely, and pessimistic, has a unit-quantity forecast, which corresponds to a plausible strategy within the market. Also, there are three unit-price forecasts, again corresponding to each of the scenarios; see Table 4.

The engineering and technology experts can estimate the nonrecurring launch cost and the recurring manufacturing cost for each unit. Again for each of the three scenarios, there is a range for launch cost, as well as a range for the unit-recurring cost target. The recurring-cost

<sup>5</sup> The relationship of unit price and quantity is described by a demand curve. For a more involved model, the marketing specialists can specify the shape of the market-demand curve and insert it into the model to help derive values for the simulation.

TABLE 4. Optimistic and pessimistic business-case scenarios.

(\$M)	Year										
	0	1	2	3	4	5	6	7	8	9	10
Optimistic											
Target unit price (\$)	40										
Target unit (240) cost (\$)											
Unit cost (\$)					41	29	24	20	18	16	15
Unit quantity					15	30	60	90	120	120	120
Revenues (\$)					600	1,200	2,400	3,600	4,800	4,800	4,800
Recurring costs (\$)					609	879	1,419	1,809	2,130	1,945	1,823
Optimistic		0	0	0	(9)	321	981	1,791	2,670	2,855	2,977
op profits (\$)											
Launch cost (\$)		0	0	(1,500)							
R&D expenses (\$)	(100)										
Pessimistic											
Target unit price (\$)	30										
Target unit (240) cost (\$)											
Unit cost (\$)					52	37	31	27	25	24	23
Unit quantity					15	30	45	45	45	45	45
Revenues (\$)					450	900	1,350	1,350	1,350	1,350	1,350
Recurring costs (\$)					778	1,124	1,394	1,236	1,142	1,077	1,028
Pessimistic		0	0	0	(328)	(224)	(44)	114	208	273	322
op profits (\$)											
Launch cost (\$)		0	0	(2,500)							
R&D expenses (\$)	(100)										

TABLE 5. Three business-case scenarios structured for Monte Carlo simulation.

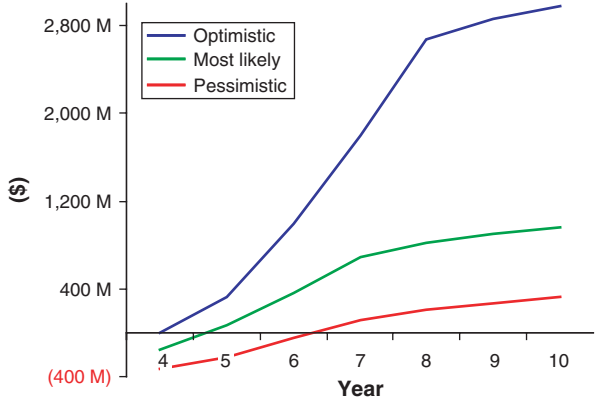
(\$M)	Year										
	0	1	2	3	4	5	6	7	8	9	10
Operating profits											
Optimistic (\$)	0	0		0	(9)	321	981	1,791	2,670	2,855	2,977
Most likely (\$)	0	0		0	(151)	73	363	688	816	900	961
Pessimistic (\$)	0	0		0	(328)	(224)	(44)	114	208	273	322
Launch costs											
Optimistic (\$)	0	0	(1,500)								
Most likely (\$)	0	0	(2,000)								
Pessimistic (\$)	0	0	(2,500)								
R&D expenses (\$)	(100)										

estimate is made for a targeted production unit, typically one in which the manufacturing process is fairly mature. The recurring-cost decreases for each unit produced, following a well-understood manufacturing learning curve.<sup>6</sup> The cost for the remaining units is then projected from the estimated cost target in the learning-curve calculation that reflects the rate of cost reduction.

Quantifying the difference between the unit price and unit cost and multiplying by the unit quantity in the optimistic, most likely, and pessimistic scenarios results in three operating-

<sup>6</sup> The formula used for the unit recurring cost learning curve is  $y = Cx^b$ .  $C$  is the estimated cost of Unit #1: about \$72 M in the most likely case example.  $x$  is the number of units.  $b$  takes the form of  $(\text{LOG}(0.85)/\text{LOG}(2))$ , where 0.85 is a typical rate of learning. We assume in this case that the target unit cost corresponds to the cost of producing the 240th unit ( $y = \$20$  M). The formula can be interpreted as each doubling of unit quantity decreases cost by 15%.

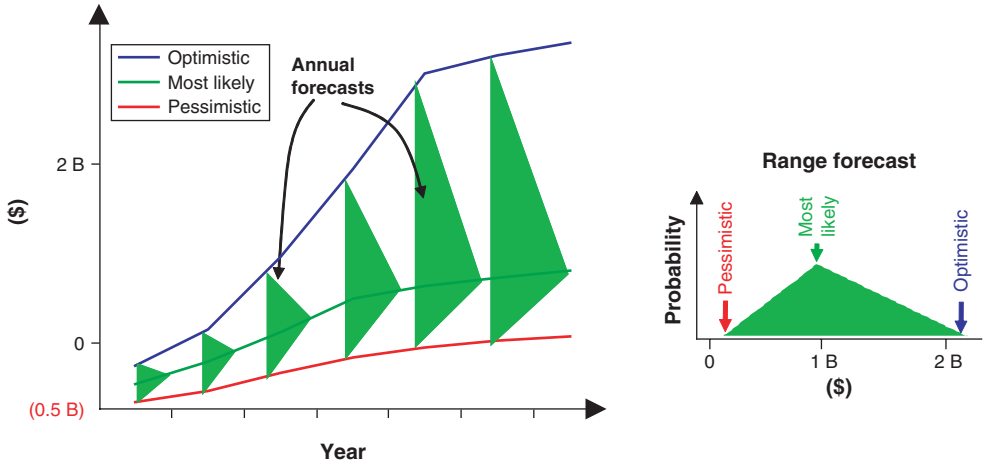
FIGURE 2. Air-freighter operating-profit scenarios.



profit cash-flow estimates for each year of the business case; see Table 5. The three operating-profit cash-flow estimates span nearly the entire range of plausible outcomes for each of the forecast years, encompassing a one-in-ten probability on the pessimistic, or low, forecast and a one-in-ten probability on the optimistic, or high, forecast; see Figure 2.

The three estimates for each year can be viewed as representing the corners of a triangular distribution that reflects a range of forecasts and thus serves as a proxy for the variation of the annual operating cash-flow forecasts at the launch date,<sup>7</sup> as can be seen in Figure 3. Using Monte Carlo software, it is relatively straightforward to construct a range forecast triangular distribution for each year of the operating profit forecast.

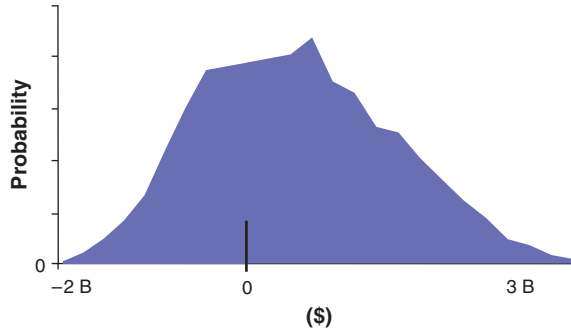
FIGURE 3. Fitting range forecast distributions to scenarios.



Using the corporate hurdle or project discount rate of 15% and applying at Year 3 the NPV function together with the Monte Carlo simulation of the multisenario operating profit yields a frequency distribution of the total range of value of the air freighter operating

<sup>7</sup> Distributions other than triangular can be used. Most risk distributions are skewed, including the triangular distributions used in the case. A skewed distribution captures the risky project concept of a low likelihood but high consequence phenomenon. A lognormal distribution, used in formal options theory, is a type of skewed distribution, but its defining parameters, such as mean and standard deviation, are more difficult to determine in the context of standard engineering and business practices. The easily comprehensible Max, Most Likely, and Min parameters that define a skewed triangular distribution can more or less approximate the more formal lognormal distribution without material impact on analytical results.

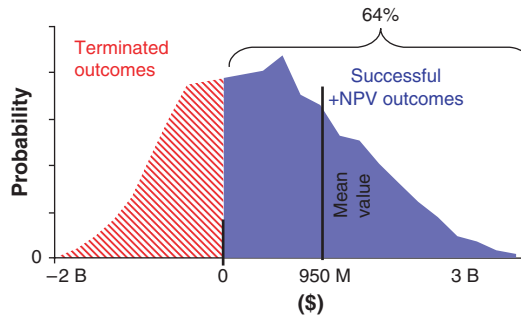
FIGURE 4. Net profits present value distribution forecast at Year 3.



profits “present valued to Year 3.” The forecast of the Year 3 launch cost also includes a range because of uncertainty of the many engineering and technology issues that comprise the launch cost (see Appendix IV). The Year 3 net profits are calculated by taking the difference of the operating profits and the launch cost even though both are distributions.<sup>8</sup> The Monte Carlo simulation provides the functionality to carry out the calculation by taking successive draws from the operating-profit and launch-cost distributions. The resulting difference calculated at each trial is the net profit for a single scenario instance. A complete simulation of hundreds of scenario trials creates a net profit present value distribution at Year 3; see Figure 4.

Year 3 is a critical decision point when the discretionary, substantial, and irreversible nonrecurring cost must be invested in order to launch the project. Only by committing the substantial launch cost investment will the corporation be able to produce the air freighter and obtain the resulting operating profits. The investment decision is irreversible because once the launch cost has been expended, those funds cannot be retrieved. The investment is discretionary because there is the option of either expending the launch cost and proceeding with the air-freighter project, termed *exercising the option*, or not expending the launch cost and terminating the project, termed *abandoning the option*.

FIGURE 5. Year 3 rational decision forecast.



Of course, the corporation will expend the launch cost if and only if at the time of the decision, Year 3, the present value of the operating profits exceeds the launch cost. In other words, if at Year 3 the forecasted discounted cash flows indicate a positive NPV, then it would be rational to invest the launch cost and initiate the project. It is important to note that each scenario in the simulation is treated as a potential cash-flow forecast. Based on these forecasts, the present value distribution at Year 3 indicates there is a 64% probability of a positive net profit, and a resulting expected (or mean) net profit of \$950 M; see Figure 5.

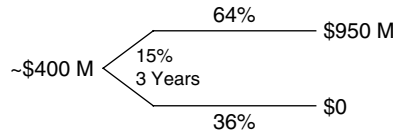
<sup>8</sup> The launch-cost-range estimate does not require discounting because it is already estimated at Year 3 value.

For the remaining 36% of the scenarios it is not financially rational to invest the substantial launch costs because of anticipated negative net profits, a loss for the corporation. Therefore, the corporation would choose not to expend the launch costs, effectively abandoning the air-freighter project, and resulting launch-cost and operating-profits cash flows would be zero.

### 3. An Investment Decision: Using Decision Trees to Capture Flexibility

Decision trees were developed to address some of the shortcomings of the NPV approach by providing a method to realize the value of flexibility within a project. To create a decision-tree valuation for the air freighter begin by analyzing the Year 3 net profit distribution derived by the what-if multiscenario approach above.<sup>9</sup> The simplest decision tree is constructed with two branches illustrating the decision outcomes at Year 3 and then discounting the results to Year 0. Discounting the net profit at 15% to Year 0 appears to value the project at around \$400 million; see Figure 6.

FIGURE 6. Decision-tree valuation technique.



Unfortunately, using the expected value of the decision-tree outcomes leads to a project valuation that is normally too high. This is because the expected value will often implicitly account for too little risk aversion. In other words, if appropriate firm decision making reflects risk aversion, a decision tree using expected values would have you overpay for the air freighter project.

To see this point in a slightly different way, consider that the decision tree is constructed such that it captures the net profit investment perspective at Year 3. However, decision tree analysis incorrectly commingles the operating profits and the launch costs when net cash flows are discounted back to Year 0. Discounting these cash flows by the same rate is almost always incorrect. The correct discount rate for a cash-flow stream should reflect the riskiness of the individual stream being discounted. And in most projects, the launch costs and the operating profits will have very different risk levels. In this case, there is no easy way to calculate a single, appropriate risk-adjusted rate for the net cash flow. Therefore, decision-tree analysis using expected values and a common discount rate does not work well for project valuation.<sup>10</sup>

In Boeing's DM method, an adjustment for risk aversion is included by using a different discount rate for the launch cost outflow and the prospective operating cash inflows. The operating profits are discounted by the market rate (15% in this case); the launch cost is discounted by the corporate bond rate (perhaps 5%) to reflect its relatively lower risk. This adjustment for risk aversion can be ratcheted up or down by changing the differential between the discount rates used for the two flows.

Some final comments about decision trees and a related older real option technique called *binomial lattices*. Although they provide a simple graphical representation of project

<sup>9</sup> It is not possible to accurately construct a decision tree directly from the initial information of the three scenarios. Though the information appears to be linear (probabilities and profit values), in actuality the total impact of the correlated, skewed, and the occasional nonlinear relationships are only revealed by the combinatorial effects of the Monte Carlo simulation.

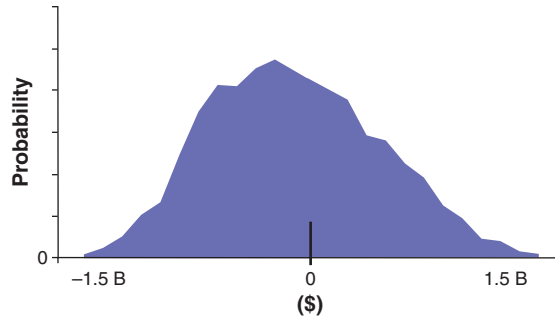
<sup>10</sup> Although it is possible to determine the appropriate risk-adjusted discount rate, it involves certainty-equivalent or risk-neutral probabilities, which are not easy to calculate.

decisions and uncertainty, decision trees and lattices have numerous shortcomings when modeling any realistic business case. Neither trees nor lattices are well accommodated in spreadsheets, the industry standard for business-case models. Most business cases involve dozens, and occasionally hundreds of variables, with uncertainties such as recurring and nonrecurring costs, schedule, technology readiness, market demand, production rates, and competitive threats. In attempting to represent all these branch uncertainties, decision trees quickly become unmanageable and also have difficulties in representing cross-influences, the so-called *joint probabilities*, among the branches. Finally, assigning the many and varied branch probabilities (or  $u, d$  parameters in a binomial lattice) is highly problematic within a business context requiring transparency and intuitiveness. In actuality, a properly structured spreadsheet-based business case with embedded Monte Carlo simulation adequately recreates the “bushy” branching of a decision tree or lattice (each trial generates a branch or path) although providing substantially more functionality as well as a bridge to system-engineering risk modeling.

#### 4. An Investment Decision: Using Real Options for Flexibility and Risk Aversion

Real options can calculate the fair value for the air-freighter project given flexibility in project planning and investment risk aversion. A real-option approach will more appropriately value a risky project by accounting for the risk aversion of the potential loss of an up-front investment. The real-option approach discounts to Year 0 the operating-profit cash flows at the hurdle rate, and the launch-cost cash flow at the investment rate.<sup>11</sup> The net profit, the difference of the two discounted cash flows, is simulated and calculated at Year 0. The result is the Year 0 net profit present-value distribution for the hundreds of cash-flow scenario trials as can be seen in Figure 7.

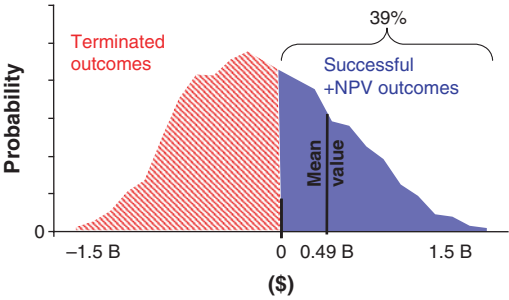
FIGURE 7. Year 0 net profit present-value distribution forecast.



The project manager needs to be financially rational on a risk-adjusted basis for the investments in the air-freighter project; see Figure 8. The solid shaded section on the right tail of the present-value distribution corresponds to the 39% probability of a successful, pos-

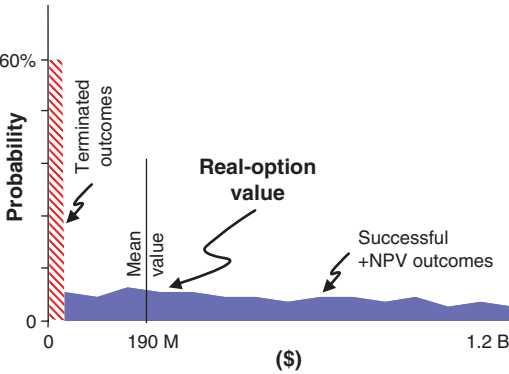
<sup>11</sup> Within Boeing, the corporate bond term rate is used in option valuation to discount the so-called private risk, or internal diversified risk. Applying a bond rate, as opposed to the more standard risk-free rate for option pricing, results in a valuation approximation that has little material impact on the final decision-making process, although appealing to management intuition. Here the low rate, the least expensive source of capital, can be understood as the resulting benefit of a diversified portfolio effect of a general obligation corporate bond. One view of real options is that it provides a “corporate-based” valuation that contrasts the value of prospective risky-project operating profits against paying off corporate bond holders. By applying an observable discount rate, the real-options business case is grounded in the realities of the capital markets. Thus, the resulting profit and loss calculations are placed on a par with how shareholders might perceive the value of the same business opportunity—a compelling argument for senior management. For illustration purposes, the risk-free rate can be used to derive a “market-based” valuation of the option. To read more about market- and private-risk dual-discount approach, see Mello and Pyo [6]. See also <http://investmentscience.com/Content/newsArticles/news3.html>.

FIGURE 8. Year 0 risk-averse rational-decision distribution.



itive NPV forecast in which the discounted operating profits exceed the launch cost.<sup>12</sup> The cross-hatched shaded section on the left tail of the present-value distribution represents the probability of those trials in which the discounted launch costs are anticipated to exceed the operating profits. Being risk averse in these cases, the project manager would rationally choose to avoid the loss by terminating the project, and there would be zero launch cost or operating-profit cash flows. Essentially, the project manager enhances the total value of the project by taking the appropriate action contingent on information revealed during the pre-launch period.

FIGURE 9. Year 0 payoff distribution.



The real-option value can be understood simply as the average net profit appropriately discounted to Year 0, the date of the initial R&D investment decision, contingent on terminating the project if a loss is forecast at the future launch decision date. This payoff calculation also has a distribution; see Figure 9. The payoff distribution illustrates the 61% probability of plausible scenario trials that are terminated with zero cash flow, whereas the remaining successful cases yield a range of expected net profits. The average value of this payoff distribution is the real-option value, approximately \$190 M in this example. The real-option value is the best estimate today of the discounted future expected net profit, conditional on risk-averse rational decision making at the time of launch.

<sup>12</sup> Here we are estimating the probability of a successful outcome of the project, meaning a risk-adjusted positive NPV value given the uncertainty of the launch-investment decision. A real-option valuation does not preclude that conditions at launch time may change necessitating a re-valuation of the prospective project profitability, nor that the launch decision will be financially risk free. A real-option valuation calculates the risk-adjusted probability of a positive NPV at launch time—a rational, but not risk-free financial decision. Exercising a real option on a project nearly always exposes the owner to a tactical decision of whether to invest the significant launch costs in the risky underlying project asset. On the other hand for financial options exercising an in-the-money call and simultaneously selling the equivalent shares of stock for a cash settlement eliminates tactical risk of owning the stock.

The formal calculation of the real-option value is done using the DM method. The spreadsheet DM method formula is as follows:

$$\text{Real-Option Value} = \text{Mean}[\text{MAX}(\overline{\text{Operating Profits}} - \overline{\text{Launch Cost}}, 0)]^{13}$$

The formula, which is a combination of Excel and Monte Carlo functionality, captures the intuition described above. The “operating profits” and “launch cost” are the present-value distributions. The payoff distribution is created by simulating several hundred scenario trails, and calculating the MAX value, with a zero threshold for terminated projects representing no cash flow. The option value, \$190 M, equals the mean value of this payoff distribution.

A real-options approach provides justification for contingent strategic R&D investments. Strategic investments allocate resources in advance of an anticipated use. The air-freighter project has a strategic value of \$190M three years prior to launch. Project management can justify investments up to this amount in technology, engineering, and marketing R&D in advance of the launch. In this case, the project engineers and market analysts need \$100 M in R&D funds today. Because the real-option value exceeds the initial R&D expense request, the project manager should approve the R&D portion of the project and fund this initial effort; see Table 6.<sup>14</sup> These funds will enable the engineers to advance the necessary technology to a state of readiness in preparation for, and effectively reducing the uncertainty of, the launch decision. Marketing analysts will be able to survey customer interest in the air freighter to gauge market interest further reducing uncertainty. Contrast this result with that of the NPV approach which would terminate the project even before initiating the R&D effort. The NPV approach fails to justify strategic investments, which leave us unprepared should the plausible but lower probability market opportunity actually materialize.

There is an intuitive understanding of real options, which is useful during those multi-scenario strategy discussions. An estimator for the real-option value can be expressed as a function of successful launch outcomes in the following formula:

$$\begin{aligned} \text{Real-Option Value} &= \text{Risk-Adjusted Success Probability} \\ &\quad \times (\text{Operating Profits} - \text{Launch Costs}) \end{aligned}$$

For example, in Figure 8 the risk-adjusted probability of success is 39%, and the appropriately discounted mean net profits value (operating profits – launch costs) of the successful outcomes is ~\$0.49 B. Using these values in the above formula produces a real-option value of the project given its contingencies:

$$\text{Real-Option Value} = 39\% \times (\$0.49 \text{ B}) \approx \$190 \text{ M}$$

## 5. Concluding Thoughts

Much of the value of real options resides not in the actual calculation of the option value, but rather in what is termed “real-options thinking.” Because options are a critical but as yet not well articulated way of how project managers conduct their decision-making process and implement project planning, applying real-options thinking provides a welcome structure to scenario discussions. The real-options planning, approach contrasts with that of NPV-driven planning, which tends to commit large dollar amounts up front to a single course of action.

The value of a real option-driven approach to project planning is tied to two key factors—an initial investment directed to uncertainty reduction followed by a contingency-based

<sup>13</sup> The overscore bar in the equation represents a distribution—formally a random variable—of the discounted cash flows at time 0.

<sup>14</sup> The air-freighter project option can be purchased for \$90 M less than its real-option value, a good deal for the shareholders. The engineers’ ability to solve aviation challenges with a high degree of efficiency is a competitive advantage, which allows the corporation to “buy” the air-freighter option below market value.



TABLE 6. Air-freighter project value using real options.

Discount rate assumptions		Present value distributions
Project risk rate	15.0%	
Investment rate	5.0%	
DM real-option calculations (\$M)		
PV <sub>0</sub> operating profits	\$734	← @ 15%
PV <sub>0</sub> launch costs	(\$1,728)	← @ 5%
Project payoff	\$0	
Project option value	\$190	
R&D expenses	(\$100)	
Total project RO value	\$90	

course of action. The first factor in the real-options-driven planning is targeting the small (relative to the launch cost), risk-averse investment toward engineering and marketing initiatives that reduce uncertainty prior to the launch commitment. The net result is at the downstream decision point of the irreversible launch investment, the project manager will be able to better determine which project scenario (optimistic, most likely, or pessimistic) is being born out in reality. Once the path or scenario is identified, there is a contingent course of action associated with the scenario that preserves the original intent of the option valuation; see Table 7. If the pessimistic scenario is actualized, then conserve the substantial launch-cost investment, terminate the project immediately, and perhaps sell off any derived patented assets. If it is the optimistic scenario, then invest the launch cost and garner the expected operating profits. If the actuality is the most likely scenario, then it may be worthwhile to delay the launch, perhaps invest additional R&D funds to preserve the opportunity and attempt to reduce the uncertainty further in order to make a clear decision at a later date.

Distinguishing between strategic and tactical investment decisions provides further rationale for using real options. Strategic decisions involve an investment commitment that is risky because the benefits are uncertain and because resources are allocated and information acquired ahead of the decision. Having the option to cancel the project if warranted significantly reduces the corporate exposure to the launch investment decision risk. This risk-lowering practice enables companies to take on smaller, higher-risk but potentially higher-return projects while maintaining fiscal responsibility. In comparison, tactical decisions are made by fully committing whatever resources and information are on hand at the decision moment. The launch commitment at Year 3 is a tactical decision, where the substantial investment is irreversible and—without the benefit of prior R&D—largely uncertain. A simple NPV analysis is typically used to help make tactical decisions, but generally is not appropriate for strategic decisions that involve phased investments made under uncertainty.

Real-options methods work for strategic decisions because of their ability to simplify and manage complex investment problems. It is generally not possible to know all of the potential factors that might affect the outcome of such investment. But it is sufficient in an uncertain environment to bound the problem, yet still remain confident in the decision-making process. By acquiring the initial resources and information necessary for informed decisions, real options allows us to “prune” possible bad outcomes and concentrate scarce

TABLE 7. Course of action at launch date contingent on scenario outlook.

Scenario	Contingent course of action at Year 3
Pessimistic	Terminate program
Most likely	Delay launch, additional R&D investment
Optimistic	Launch immediately, receive operating profits

investment resources on those truly promising opportunities. The DM method simplifies the calculation behind this strategy.

## Business Engineering

Historically, business managers tend to throw their needs, including financial goals, “over the wall” to systems engineering in the form of a requirements document. However, what is really needed, is a working relationship between business and systems engineering. Technology, products, and services are best designed with a strong understanding of the business process context. A concurrent business-engineering practice is one in which engineering and business work jointly to simultaneously design both the business process and the technology solutions to support it. Such a practice enables a dialogue around and trade-off between business and technology considerations, fostering greater levels of innovation and optimization. Business engineering applies advanced investment and risk modeling and simulation technologies to trades of system-engineering performance and business objectives to support key strategic decisions (see Appendix IV).

Business engineering helps determine optimal solutions to system performance, cost-effective product design and production, and business objectives. Risks are elicited by simulation trade-study models, which provide project management with key insights to control identified risk drivers in a staged process of project development. The methods and tools used in business engineering, algorithms such as uncertainty modeling and real options, provide a mathematical foundation for a scientific and “engineering-like” approach to risk identification and quantification of impact. The methods enhance the level of engineering managers’ confidence in risk reduction through targeted allocation of risk mitigation and reserve funds. The resulting approach closely reflects how savvy project managers already flexibly balance technology and cost, and schedule risks and opportunities.

## Appendix I: DM Method Extensions

The simplest DM method extension is the conversion to an Excel logic function.

$$\text{Real-Option Value} = \text{Mean}\{\text{if}[(\overline{\text{Operating Profits}} - \overline{\text{Launch Cost}}) > 0, \\ (\overline{\text{Operating Profits}} - \overline{\text{Launch Cost}}), 0]\}$$

Part of the advantage of the logic formula is improved clarity of the real-option valuation. Additional advantages can accrue to business analysts that would prefer to capture the intuitive appeal of logic for strategic business decision but also realize that “operating profits” and “launch cost” could be modeled by fairly complex spreadsheet scenarios. For example, operating profit volatility can be more accurately modeled by integrating a model of a dynamic demand curve and production uncertainty.

There are additional options within the DM method framework. For example, launch cost, which in the example above is a range value (a type of option termed *variable strike*) can also be a fixed cost. A fixed cost (termed a *strike price*) is the most common type of financial option. Another example is an exit option either to license or sell the technology developed, e.g., for \$50 M, in the event the project is terminated. The value of the terminated, unsuccessful project is therefore \$50 M, not \$0. Combining these two options, the spreadsheet formula for the complex project option becomes:

$$\text{Real-Option Value} = \text{Mean}[\text{MAX}(\overline{\text{Operating Profits}} - \overline{\text{Launch Cost}}, 50)]$$

A project type that frequently arises at Boeing is a fixed-price government-project bid where the uncertainty is the one-time cost of the system. In this case, the traditional option

variables are reversed with the benefit being a fixed value. The DM method is able to calculate the option value of the bid opportunity.

$$\text{Real-Option Bid Value} = \text{Mean}[\text{MAX}(\text{Bid Price} - \overline{\text{System Costs}}, 0)]$$

The option types above are termed *call options*, meaning investing in an opportunity that potentially will pay off if there is an “upside” increase in value. There is another option type termed the *put option*, which is an option that will potentially pay off if there is a “downside” turn of events. Insurance is a type of put option, where in the event of a loss, a policy holder receives a pay out from his/her insurance company. Another type of put option commonly used in a business environment is a service guarantee, such as customer service agreements (CSA), or, for expensive leased assets such as cars and airplanes, a residual value guarantee (RVG). Put options often are used in contingent clauses in contracts to tailor the value to the performance risks of the contract. Put options are also the basis for hedging strategies in which an option investment premium is paid to purchase a guaranteed protection level (a “deductible” in consumer insurance) in the event of a worst-case loss outcome. The DM method is easily extended to value a put option as follows:

$$\text{Real-Put-Option Value} = \text{Mean}[\text{MAX}(\text{Guarantee Cost} - \overline{\text{Loss}}, 0)]$$

## Appendix II: Comparing the Dathar-Mathes Method and the Black-Scholes Formula

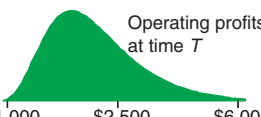
The DM method can be shown to be mathematically equivalent to the Black-Scholes formula given certain assumptions. These two are mechanically different representations of the same underlying economics. Table A.1 illustrates a very simplified, but typical discounted cash flow (or NPV) analysis set up. The DM method<sup>15</sup> uses the distribution of forecast

TABLE A.1. Black-Scholes compared to Datar-Mathews method.

(\$M)	Year	0	1	2	3	...10
Operating profits					\$2,500	
Investment					\$2,000)	

Time (t)  
Risk-free rate ( $R_f$ )  
Discount rate ( $R_r$ )

3  
5.0%  
15.0%



Operating profits  
at time T

Operating profits  
Mean  
StdDev

\$2,500  
\$1,000

Black-Scholes formula		
PV <sub>0</sub> Asset (S)	\$1,594.07	=EXP(-Rr*t)*(Mean)
PV <sub>0</sub> Exercise (X)	(\$1,721.42)	=EXP(-Rf*t)*Investment
Sigma	22.2%	=SQRT(LN(1+(StdDev/Mean)^2))/SQRT(t)
Option value	\$194.50	=Nd <sub>-1</sub> *S-(Nd <sub>-2</sub> *(-X))
d <sub>1</sub>	-0.01	=(LN(S/(-Investment)))+(Rf+0.5*(Sigma^2))*t)/(Sigma*SQRT(t))
N(d <sub>1</sub> )	0.50	=NORMSDIST(d <sub>1</sub> )
N(d <sub>2</sub> )	0.35	=NORMSDIST(d <sub>1</sub> -(Sigma*SQRT(t)))

D-M method		
PV <sub>0</sub> Asset (A)	\$1,594.07	=EXP(-Rr*t)*OpProfits
PV <sub>0</sub> Exercise (X)	(\$1,721.42)	=EXP(-Rf*t)*Investment
Payoff	\$0.00	=MAX(A+X,0) or IF((A>-X),A+X,0)
Option value	\$194.50	=Average(Payoff)

<sup>15</sup> Datar-Mathews formula:  $C_0 = E[e^{-\mu t} \bar{S} - e^{-r t} X]^+$ , an expectation formula where  $\bar{S}$  is the random variable for operating profits,  $\mu$  and  $r$  are the risky-asset and the risk-free discount rates respectively, and  $+$  is the MAX function. The simulation for the DM method is typically run for 10–20,000 trials as it gradually converges on the Black-Scholes value.

cash flows to calculate the option value, whereas Black-Scholes<sup>16</sup> uses a value of sigma  $\sigma$ . Further, the DM method implicitly adjusts the discount rate to account for the underlying risk. The option value is easily understood as expected pay-off resulting from rational exercise decisions. The DM method provides a better estimate of option value when the strict theoretical assumptions of Black-Scholes are compromised in real life. For example, the DM method can easily deal with triangular (non-lognormal) cash-flow distributions and random exercise price.

**Appendix III: Comparing Financial Options and Real Options**

The history of options is quite colorful and surprisingly a lot longer than most people think (Chance [2]). Although it is not known exactly when the first option contract traded, the Romans and Phoenicians used contracts similar to options in shipping. In Holland, trading in tulip options blossomed during the early 1600s. In 1670, the Royal Exchange in London became the first exchange for trading options. In the United States, the Chicago Board of Trade (CBOE) began exchange trading in 1848. CBOE estimates that in 2006 it traded more than \$15 trillion in options, a value larger than the U.S. economy. Fischer Black and Myron Scholes first published an option pricing model in 1973. The term “real option” was coined a few years later by Professor Stewart Myers of MIT.

Real options are derived from the mathematics financial options and practices of options trading in the options markets. Here is a brief comparison of the variables:

Financial options	Real options
Usually exchange traded	Not usually traded
Contract with contingencies	Strategy with contingencies
Underlying is a stock	Underlying is a product/project
Premium payment	R&D investments
Strike or exercise price	Non recurring investment
“Exercise” an option	Commitment to production or bid
Time to exercise	Time to commitment
Payoff is stock or cash	Payoff is product operating profits
Variance of stock (sigma)	Variance of operating profits
Mathematics correctly prices	Mathematics correctly prices
contingent investment securities	contingent investment strategies

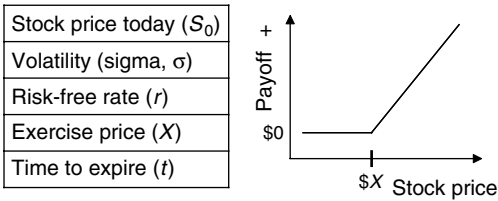
The most simple option is a so-called European call option that can be exercised only on a specified date. The option value is calculated based on five parameters, four of which are known, the exception being the volatility parameter. The option contract specifies the exercise price ( $\sim$  \$2 B for the air-freighter example) to purchase the stock. Should the stock price on the exercise date  $t$  exceed the exercise price, then the option is “in-the-money.” The option holder can either purchase the stock at the exercise price or settle for a cash payoff, the difference between the current stock price and the exercise price. If the stock price falls below the exercise price on date  $t$ , then the option expires and the payoff is \$0; see Figure A.1.

**Appendix IV: Extensions to Systems Engineering**

The launch cost can be simply modeled as a triangular distribution with Min-Most Likely-Max range values of \$1,500 M, \$2,000 M, and \$2,500 M, respectively. In reality, the launch cost is comprised of many cost elements, and each element having a range estimate. In order

<sup>16</sup> Black-Scholes formula:  $C_0 = S_0N(d_1) - Xe^{-rt}N(d_2)$ , where  $d_1 = (\ln(S_0/x) + (r_f + \sigma^2/2)t)/\sigma\sqrt{t}$  and  $d_2 = d_1 - \sigma\sqrt{t}$ .

FIGURE A.1.



to better estimate the true launch-cost range a more detailed launch “cost risk” model is constructed; see Table A.2. The estimates of the cost elements are correlated to a physical system engineering design variable (Max Take Off Weight, MTOW), which itself is a range estimate. Similarly, a MTOW detailed system “engineering risk” model could be constructed using comparable concepts, but related to performance instead of business.

TABLE A.2. Launch-cost risk estimate detail.

Nonrecurring cost/ risk estimate System engineering element	Simulation distribution	Range estimate			MTOW correlation
		Low	Most likely	High	
Max take off weight MTOW (lbs)	195,000	125,000	195,000	245,000	—
Structures	\$760,000,000	\$500,000,000	\$760,000,000	\$825,000,000	0.65
Systems	\$695,000,000	\$545,000,000	\$965,000,000	\$965,000,000	0.45
Propulsion integration	\$130,000,000	\$85,000,000	\$130,000,000	\$165,000,000	0.20
Manufacturing operations	\$190,000,000	\$100,000,000	\$190,000,000	\$245,000,000	0.20
Certification and testing	\$135,000,000	\$125,000,000	\$135,000,000	\$250,000,000	0.25
Marketing and overhead	\$90,000,000	\$50,000,000	\$90,000,000	\$125,000,000	0.10
Total nonrecurring cost/risk	\$2,000,000,000				
Cost risk mean	\$1,993,331,574				
Cost risk Stdev	\$147,314,693				

A simulation creates the launch-cost distribution; see Figure A.2. The distribution statistics, mean and standard deviation, enable us to create a launch-cost lognormal distribution that more accurately captures the launch-cost range estimate. However, the superimposed triangle approximates the simulation distribution and can be applied, e.g., to a less complex model, such as in Table 5, without material impact on analytical results.

FIGURE A.2. Launch-cost detail distribution.

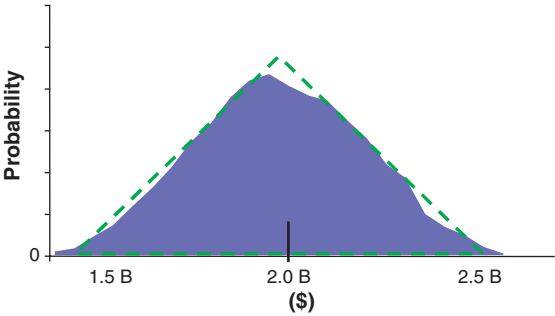
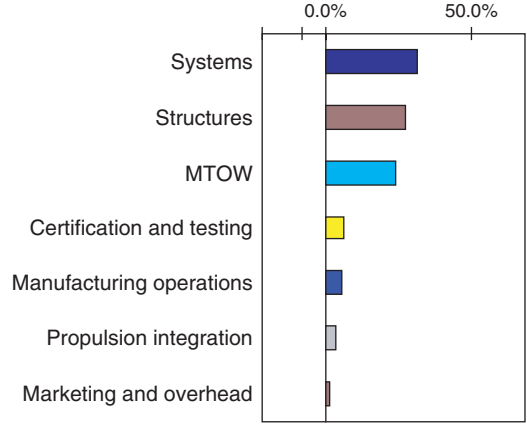


FIGURE A.3. Launch-cost elements sensitivity (contribution to variance).

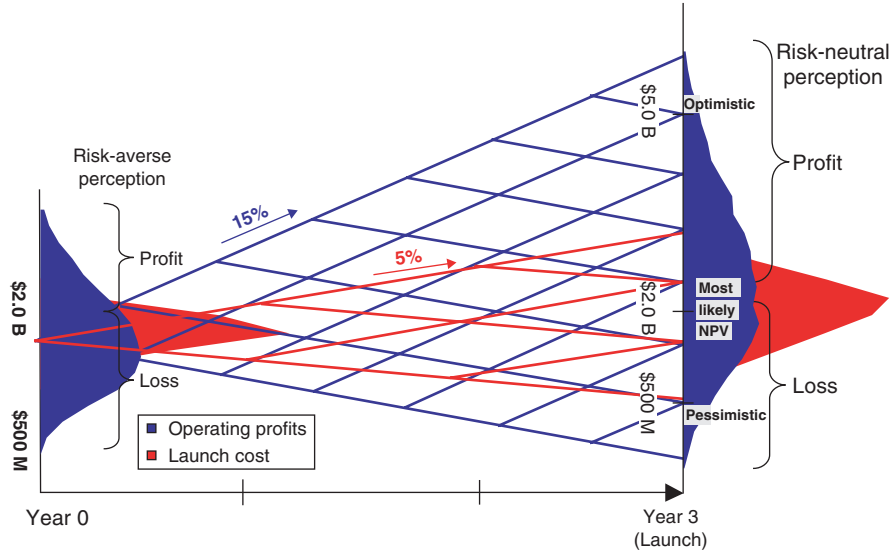


Monte Carlo simulation software provides a sensitivity analysis showing those elements that most contribute to the launch-cost variance (Figure A.3). The principal launch-cost risk drivers are systems, structures, and MTOW. Therefore, much of the \$100 M requested R&D funds ought to be directed toward reducing risk and uncertainty in these areas.

Appendix V: The Big Picture

Why do the “success” probabilities in Figure 5 and Figure 8 differ? Figure 5 illustrates the risk-neutral probabilities at Year 3, whereas Figure 8 illustrates the risk-averse probabilities at Year 0. At Year 0, the manager may be quite risk averse because \$100 M, a substantial sum of money, is to be invested well before the launch opportunity is viable. This risk aversion translates into a perceived reduction in the chances of success. The DM method implicitly adjusts the probabilities to account for risk aversion by appropriate use of differential discount rates. The intent of the \$100 M investment is to resolve a number of the project risks. With some of the uncertainties literally behind by Year 3, launch prospects can be examined in a less risky framework and will have a different perspective on the success rate. At that

FIGURE A.4. Cash flow cone of uncertainty (decision lattice) and dispersion caused by discounting.



time it can be determined whether the project meets the 15% required rate of return on the investment of the \$2 B launch cost; see Figure A.4.

One can argue that the Year 3 outlook will change because of the \$100 M investment and the resulting learning. Of course, the model simply attempts to value the project today given the best projections of the future. However, the project outlook should be updated as risks are mitigated and there is a better understanding of the market. At each funding period, or stage gate, the worthiness of the project should be updated with this new information. A more sophisticated multistage model sets expectations for risk reduction and value enhancement contingent on funding.

## References

- [1] F. Black and M. Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy* 81(3):637–654, 1973.
- [2] D. Chance. Teaching notes: A brief history of derivatives. *Financial Engineering News* 41, 2005.
- [3] V. Datar and S. Mathews. Datar-Mathews method for quantitative real option value. U.S. Patent 6862579, filed July 10, 2002, and issued March 1, 2005.
- [4] V. Datar and S. Mathews. European real options: An intuitive algorithm for the Black-Scholes formula. *Journal of Applied Finance* 14(1), 2004.
- [5] S. Mathews, V. Datar, and B. Johnson. A practical method for valuing real options: The Boeing approach. *Journal of Applied Corporate Finance* 19(2), 2007.
- [6] A. S. Mello and U. Pyo. Real options with market risks and private risks. *Journal of Applied Corporate Finance* 15(2), 2002.

## Contributing Authors

**Kenneth R. Baker** (“Safe Scheduling”) is the Nathaniel Leverone Professor of Management at the Tuck School of Business at Dartmouth College. His current teaching focuses on mathematical modeling, and his most recent textbook is entitled *Optimization Modeling with Spreadsheets* (Duxbury Press). His textbook *Elements of Sequencing and Scheduling* has been used in graduate courses for more than 30 years. He is an INFORMS fellow and a fellow of the Manufacturing and Service Operations Management Society.

**Gary M. Erickson** (“Differential Games in Marketing Science”) is a professor of marketing and GM Nameplate Faculty Fellow at the University of Washington Business School. He has been a member of INFORMS since 1978. He has been an associate editor with *Management Science* since 1987, and a member of the regular editorial board of *Marketing Science* since 2005.

**Willy Herroelen** (“Generating Robust Project Baseline Schedules”) is an emeritus professor of operations management at the Research Center for Operations Management of the Faculty of Economics and Applied Economics at Katholieke Universiteit Leuven in Belgium, where he also earned his Ph.D. in applied economics. He is co-author of the book, *Project Scheduling—A Research Handbook* (Springer). He is a former associate editor for *Management Science*, and is currently senior editor for *Production and Operations Management*. His current research focuses on proactive/reactive resource-constrained project scheduling.

**Frederick S. Hillier** (“Trends in Operations Research and Management Science Education at the Introductory Level”) is a professor of operations research, emeritus, at Stanford University. He is the author or co-author of six books, including textbooks such as *Introduction to Operations Research* (with the late Gerald J. Lieberman) and *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets* (with Mark S. Hillier). He was awarded the 2004 INFORMS Expository Writing Award for the eighth edition of *Introduction to Operations Research* and is currently preparing the ninth edition.

**Mark S. Hillier** (“Trends in Operations Research and Management Science Education at the Introductory Level”) is an associate professor at the Business School at University of Washington in the Department of Information Systems and Operations Management. His research interests include issues in component commonality, applications of mathematical programming, and the design of production systems. He has won several teaching awards for his teaching of management science. He is co-author of the textbook *Introduction to Management Science: A Modeling and Case Studies Approach with Spreadsheets* (with Frederick S. Hillier), now in its third edition.

**Michael P. Johnson** (“Community-Based Operations Research”) is an associate professor in the Department of Public Policy and Public Affairs at the University of Massachusetts Boston. His research interests lie primarily in public-sector facility location and service delivery, with applications to subsidized and affordable housing, senior services, and community corrections. His community service includes community planning and educational policy



design. He received his Ph.D. in operations research from Northwestern University in 1997 and his B.S. from Morehouse College in 1987.

**Leon S. Lasdon** (“Computational Global Optimization”) holds the David Bruton Jr. Chair in Business Decision Support Systems in the Information, Risk, and Operations Management Department at McCombs School of Business, University of Texas at Austin. He is co-author of the Microsoft Excel Solver. His OQNLP and MSNLP multistart solvers are available from the General Algebraic Modeling System and TOMLAB. His LSGRG2 nonlinear optimizer is available from Frontline Systems as part of Premium Excel Solver. He is the author or co-author of more than 120 refereed journal articles and three books (<http://utexas.edu/courses/lasdon>, link to “papers”).

**Scott Mathews** (“Business Engineering: A Practical Approach to Valuing High-Risk, High-Return Projects using Real Options”) is an associate technical fellow at The Boeing Company and is the technical lead for the Computational Finance and Stochastic Modeling team for the Modeling and Simulation section within the Boeing research and development division.

**Sigurdur Ólafsson** (“Nested Partitions Optimization”) is an associate professor in the Department of Industrial and Manufacturing Systems Engineering at Iowa State University, where he has been since receiving his Ph.D. from the University of Wisconsin–Madison in 1998. The primary focus of his research is discrete optimization. On the methodological side his work has focused on metaheuristics. He is interested in numerous application areas where discrete optimization is useful, including simulation optimization, scheduling, and data mining.

**János D. Pintér** (“Computational Global Optimization”) is an OR/MS researcher and practitioner. His professional interests are primarily related to nonlinear optimization, including algorithm/software development and applications. He has operated his own consulting service, Pintér Consulting Services (<http://www.pinterconsulting.com>), since 1994. He holds degrees in mathematics, with specializations in operations research, stochastic and global optimization. He received an M.Sc. from Eötvös University, a Ph.D. from Lomonosow University, and a D.Sc. from Hungarian Academy of Sciences. He has written and edited several books, serves on editorial boards, and authored more than 180 research papers and technical reports. He has worked and presented lectures in over 30 countries of the Americas, Europe, and the Pacific Region.

**R. Tyrrell Rockafellar** is a professor emeritus at the University of Washington (Seattle) and an adjunct research professor at the University of Florida. He graduated with a Ph.D. in mathematics from Harvard University in 1963. He is a well-known and widely published author (<http://www.ise.ufl.edu/rockafellar/>), and has been awarded several honorary prizes. These prizes include the Dantzig Prize from SIAM and Mathematics Programming Society in 1982, the Lanchester Prize from INFORMS in 1998, and the John von Neumann Theory Prize from INFORMS in 1999. His special interests include optimization theory, convex analysis, and variational analysis.

**Leyuan Shi** (“Nested Partitions Optimization”) is a professor of industrial and systems engineering at the University of Wisconsin–Madison. She received her Ph.D. in applied mathematics from Harvard University in 1992. Her research interests focuses on effective methodology for optimization of large-scale discrete systems and its practical applications in such areas as supply chain optimization and production planning and scheduling.

**Jim Salmon** (“Business Engineering: A Practical Approach to Valuing High-Risk, High-Return Projects using Real Options”) is a lead airplane design engineer in The Boeing Company’s Commercial Airplane Product Development group. He is involved in helping align Boeing’s technology investment portfolio to reflect increased market demand for advanced commercial airplanes.

**Karen Smilowitz** (“Community-Based Operations Research”) is an assistant professor in the Department of Industrial Engineering and Management Sciences at Northwestern University. She currently holds the Junior William A. Patterson Chair in Transportation. She received her Ph.D. in civil and environmental engineering from the University of California, Berkeley in 2001. Her research interests include logistics, transportation operations, and nonprofit operations research applications. She received a National Science Foundation CAREER Award in 2004, and a Sloan Industry Studies Fellowship in 2005.

**Dan Trietsch** (“Safe Scheduling”) is a professor of industrial engineering at American University of Armenia, Yerevan. His teaching interests include operations research, operations management, project management, quality management and practical application of focusing principles (an industrial project course). His current research interest is in stochastic balance principles, for example, safe scheduling. He is the author of *Statistical Quality Control: A Loss Minimization Approach* (World Scientific, Series on Applied Mathematics, Volume 10).