Jörg Kalcsics
Stefan Nickel
Editors

Operations Research Proceedings 2007

GOR
Gesellschaft für Operations Research e.V.

Springer

Operations Research Proceedings 2007

Selected Papers
of the Annual International Conference
of the German Operations Research Society (GOR)

Saarbrücken, September 5–7, 2007

Jörg Kalcsics · Stefan Nickel
(Editors)

Operations Research Proceedings 2007

Selected Papers
of the Annual International Conference
of the German Operations Research Society (GOR)

Saarbrücken, September 5–7, 2007

Springer

Dr. Jörg Kalcsics
Prof. Dr. Stefan Nickel

Saarland University
Faculty of Law and Economics
Chair of Operations Research and Logistics
P.O. Box 15 11 50
66041 Saarbrücken
Germany

j.kalcsics@orl.uni-saarland.de
s.nickel@orl.uni-saarland.de

ISBN 978-3-540-77902-5 e-ISBN 978-3-540-77903-2

DOI 10.1007/978-3-540-77903-2

Library of Congress Control Number: 2008922729

Production: le-tex Jelonek, Schmidt & Vöckler GbR, Leipzig
Cover design: WMX Design GmbH, Heidelberg

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

Preface

The symposium Operations Research 2007 was held from September 5-7, 2007 at the Saarland University in Saarbrücken. This international conference is at the same time the annual meeting of the German Operations Research Society (GOR).

The transition in Germany (and many other countries in Europe) from a production orientation to a service society combined with a continuous demographic change generated a need for intensified Operations Research activities in this area. On that account this conference has been devoted to the role of Operations Research in the service industry. The links to Operations Research are manifold and include many different topics which are particularly emphasized in scientific sections of OR 2007.

More than 420 participants from 30 countries made this event very international and successful. The program consisted of three plenary, eleven semi-plenary and more than 300 contributed presentations, which had been organized in 18 sections. During the conference, the GOR Dissertation and Diploma Prizes were awarded. We congratulate all winners, especially Professor Wolfgang Domschke from the Darmstadt University of Technology, on receiving the GOR Scientific Prize Award.

Due to a limited number of pages available for the proceedings volume and the ambition to compile only high quality papers, the length of each contribution as well as the total number of accepted papers had to be restricted. Submitted manuscripts have therefore been reviewed by the section chairs and 77 of them have been accepted for publication. These contributions represent a wide portfolio chosen from the comprehensive spectrum of Operations Research in theoretical research as well as practical experience. We would like to thank all participants

of the conference for submitting high quality manuscripts for the proceedings. Our thanks also go to the section chairs for their support in acquiring interesting contributions and their reviewing work.

Various persons and organizations contributed to the great success of the conference. We would like to thank the GOR-board and Bärbel Niedzwetzki from the GOR administrative office for the uncomplicated and constructive collaboration as well as our sponsors for their support. Our thanks also go to the members of the program and organization committees of the congress. Furthermore, we are grateful to all speakers from Germany and from all over the world for their active participation and to the section chairs as well as to the session chairs for their professional moderation of the interesting talks and discussions.

Moreover, we express our special thanks to Dipl.-Kffr. Ursula-Anna Schmidt, Dipl.-Math. oec. Sebastian Velten, Dipl.-Kfm. Hans-Peter Ziegler, and Karin Hunsicker for their excellent job before, during and after the congress. Finally, we would like to thank Stefanie Schweitzer and Lisa Scheer for their help in compiling this proceedings volume as well as Barbara Feß and Barbara Karg from Springer-Verlag for their support concerning its publication.

Saarbrücken, *Jörg Kalcsics*
December 2007 *Stefan Nickel*

Committees

Program Committee

H.W. Hamacher (TU Kaiserslautern)
R. Klein (U Augsburg)
S. Nickel (U Saarbrücken, chair)
G. Schmidt (U Saarbrücken)
T. Spengler (TU Braunschweig)
K.H. Waldmann (U Karlsruhe)
B. Werners (U Bochum)

Local Organizing Committee

N. Cwikla
J. Kalcsics
S. Nickel (chair)
U.A. Schmidt
S. Strohmeier
S. Velten
H.P. Ziegler

Scientific Sections and Section Chairs

Applied Probability and Stochastic Programming
Schultz (U Duisburg-Essen)

Artificial Intelligence, Business Intelligence and Decision Support
Fink (UBw Hamburg)

Continuous Optimization
Stein (U Karlsruhe)

Discrete and Combinatorial Optimization
Martin (TU Darmstadt)

Econometrics, Game Theory and Mathematical Economics
Eckwert (U Bielefeld)

Energy, Environment and Life Sciences
Pickl (UBw München)

Entrepreneurship and Innovation
Raith (U Magdeburg)

Finance, Banking and Insurance
Wilkens (KU Eichstätt-Ingolstadt)

Forecasting and Marketing
Küsters (KU Eichstätt-Ingolstadt)

Health Care Management
Melo (HTW Saarbrücken)

Managerial Accounting and Auditing
Friedl (TU München)

Multi Criteria Decision Making
Geldermann (U Göttingen)

Production and Service Operations Management
Tempelmeier (U Köln)

Retail, Revenue and Pricing Management
Spinler (HHL Leipzig)

Scheduling and Project Management
Pesch (U Siegen)

Simulation, System Dynamics and Dynamic Modelling
Furmans (U Karlsruhe)

Supply Chain Management and Traffic
Stadtler (U Hamburg), Haase (TU Dresden)

Software
Kalcsics (U Saarbrücken)

Contents

Part I

Dissertation Award Winners

Expected Additive Time-Separable Utility Maximizing Capacity Control in Revenue Management

Christiane Barz

Graduate School of Business, University of Chicago
christiane.barz@gsbchicago.edu

Summary. We briefly discuss the static capacity control problem from the perspective of an expected utility maximizing decision-maker with an additive time-separable utility function. Differences to the expected revenue maximizing case are demonstrated by means of an example.

Within the last two decades, revenue management – the control of product availability and pricing decisions in order to maximize revenue – has spread in both theory and practice. More and more industries adopted revenue management practices, new models have been studied extensively, structural properties of the optimal control policy have been proven, heuristics have been promoted, and various extensions and alternatives have been suggested (for a review see [5]). Even in the latest literature on revenue management, however, expected revenue is the most widespread optimality criterion in use.

The assumption that different alternatives are evaluated solely by expected values is a standard assumption and justified e.g. if the company repeats the same decision problem over thousands of instances. But today more and more industries with sometimes only very few repetitions are adopting revenue management practices. In applications, where a single poor realization can have a major impact on the financial condition of the business, risk-averse approaches might need to be considered. Failing to suggest mechanisms for reducing unfavorable revenue levels, traditional risk-neutral revenue management models fall short of meeting the needs of a risk-averse planner.

Since the assumption of an additive time-separable utility function is the one most frequently used in combination with Markov decision processes, we will briefly discuss the implications of this preference struc-

ture on an optimal capacity control policy. We start with a brief review of the static capacity control problem in Sect. 1. Then, we reformulate this decision problem from the perspective of a (risk-averse) expected utility maximizing decision maker in Sect. 2. The effect of this new perspective on an optimal decision policy is demonstrated by means of an example. In particular, we critically discuss the special structure of temporal and risk preferences imposed in Sect. 3. For simplicity, we will stick to the terminology of the airline industry throughout.

The papers by [6] and [2] also introduce risk-aversion into the capacity control formulation and are hence closest to our analysis. The underlying preference structures are, however, different from our approach. See [1] for a more comprehensive literature review and an in-depth discussion of optimal capacity control from the perspective of an expected utility maximizing decision-maker.

1 The Static Capacity Control Model

The very basic single-leg static capacity control model considers a non-stop flight of an airplane with a capacity of C seats that is to depart after a certain time. There are $i_{\max}$ ($i_{\max} \in \mathbb{N}$) booking classes with positive fares ordered according to $\rho_1 \geq \rho_2 \geq \cdots \geq \rho_{i_{\max}}$. Demand for the different booking classes arrives in a strict low-to-high fare order. Total demands for the booking classes $1, \ldots, i_{\max}$ are assumed to be independent discrete random variables $D_1, \ldots, D_{i_{\max}}$ with outcomes $d \in \{0, \ldots, d_{\max}\}$. Neither cancelations nor no-shows are allowed.

At the time the total demand of a booking class is known, the decision-maker has to determine the amount of demand to be accepted, i.e. the number of seats that should be sold. Traditionally, the decision-maker aims at maximizing the expected revenue of a flight.

1.1 The Underlying MDP

The objective of finding a policy maximizing the expected revenue can be reduced to solving the optimality equation of a finite stage Markov decision model $MDP(i_{\max}, \mathfrak{X}, \mathfrak{A}, p_i, r_i, V_0)$ with planning horizon $i_{\max}$, state space $\mathfrak{X} = \{(c,d) \in \mathbb{Z} \times \mathbb{N}_0 \mid c \leq C,\ d \leq d_{\max}\}$, where we refer to c as the remaining capacity and to d as the demand observed for the actual booking class, set $A(c,d) = \{0, \ldots, d\}$ of admissible actions in state (c,d), transition law p_i for $i = i_{\max}, i_{\max}-1, \ldots, 1$ such that $p_i((c,d), a, (c-a, d')) = P(D_{i-1} = d')$ and 0 otherwise, one-stage rewards $r_i((c,d), a) = a\rho_i$, and terminal reward $V_0((c,d)) = 0$ for $c \geq 0$ and $V_0((c,d)) = \bar{\rho}c$ for $c < 0$ with $\bar{\rho} > \max_i\{\rho_i\}$.

A (Markov) policy $\pi = (f_{i_{max}}, f_{i_{max}-1}, \ldots, f_1)$ is then defined as a sequence of decision rules f_i specifying the action $a = f_i(c,d)$ to be taken at stage i in state (c,d). Let $\mathit{\Pi}$ denote the set of all policies and $(X_{i_{max}}, X_{i_{max}-1}, \ldots, X_0)$ the state process of the MDP. In addition, introduce for all $(c,d) \in \mathfrak{X}$

$$V^*(c,d) = \max_{\pi \in \mathit{\Pi}} E_\pi \left[\sum_{i=1}^{i_{max}} r_i(X_i, f_i(X_i)) + V_0(X_0) \mid X_{i_{max}} = (c,d) \right] \quad (1)$$

to be the maximum expected revenue.

For all booking classes i, given the residual capacity c and demand d, the decision-maker is interested in the number $a = f_i(c,d) \in \{0, \ldots, d\}$ of seats that should be sold in order to achieve the maximum expected revenue V^*.

It is well-known that $V^* \equiv V_{i_{\max}}$ is the unique solution to the optimality equation

$$V_i(c,d) = \max_{a=0,\ldots,d} \left\{ a\rho_i + \sum_{d'=0}^{d_{\max}} P(D_{i-1} = d') V_{i-1}(c-a, d') \right\}, \quad (2)$$

which can be obtained for $i = 1, \ldots, i_{\max}$ by backward induction starting with V_0. Every policy π^* that is formed by actions $a^* = f_i^*(c,d)$ each maximizing the right hand side of (2) is optimal, i.e. leads to V^*.

1.2 Structural Results

Many authors have shown that the structure of an optimal policy for this static model is as follows (see e.g. [5, pp. 36–40]):

Theorem 1. *For the static problem there exists an optimal policy* $\pi^* = (f^*_{i_{\max}}, f^*_{i_{\max}-1}, \ldots, f^*_1)$ *such that*

$$f_i^*(c,d) = \begin{cases} \min\{d, c - y^*_{i-1}\} & c > y^*_{i-1} \\ 0 & c \le y^*_{i-1} \end{cases},$$

with protection levels $y^*_{i-1} = \max\{c \in \{0, \ldots, (i-1)d_{\max}\} : \rho_i < \sum_{d'=0}^{d_{\max}} P(D_{i-1} = d') \left[V_{i-1}(c) - V_{i-1}(c-1)\right]\}$. *Optimal protection levels are increasing in* i, *i.e.* $y^*_{i_{\max}-1} \ge y^*_{i_{\max}-2} \ge \ldots \ge y^*_1 \ge y^*_0 = 0$.

Given d requests from customer class $i = 1, \ldots, i_{\max}$, a non-negative number of y^*_{i-1} seats (the so-called protection level of class $i-1$) is reserved for future demand of classes $i-1, \ldots, 1$. The protection levels y^*_{i-1} are lower for higher value demand.

2 Maximizing Additive Time-Separable Expected Utility

In his definition of Markov decision processes, Rust explicitly assumes that the decision-maker has a utility function that is additively separable, since other structures are "computationally intractable for solving all but the smallest problems"[4, p. 630].

So assume that the one-stage rewards $r_i(X_i, f_i(X_i))$ and the terminal reward $V_0(X_0)$ are transformed by increasing and concave von Neumann-Morgenstern utility functions u_i and u_0 with $u_i(0) = u_0(0) = 0$. We refer to the textbook of [3] for a general introduction to expected utility theory.

2.1 The Underlying MDP

Suppose that the decision-maker is interested in finding

$$V^{u*}(c,d) = \max_{\pi \in \Pi} E_\pi \left[\sum_{i=1}^{i_{\max}} u_i(r_i(X_i, f_i(X_i))) + u_0(V_0(X_0)) | X_{i_{\max}} = (c,d) \right],$$

the maximum expected (additive time-separable) utility starting with capacity c given d requests from class $i_{\max}$ over all policies $\pi \in \Pi$.

Then, $V^{u*} \equiv V^u_{i_{\max}}$ is the unique solution of the optimality equations

$$V^u_i(c,d) = \max_{a=0,\dots,d} \left\{ u_i(a\rho_i) + \sum_{d'=0}^{d_{\max}} P(D_{i-1} = d') V^u_{i-1}(c-a,d) \right\} \quad (3)$$

with terminal reward $V^u_0(c,d) = u_0(V_0(c,d)) = 0$ for all $c \geq 0$ and $V^u_0(c,d) = u_0(V_0(c,d)) = c\bar{\rho}^u$ for all $c < 0$, where $\bar{\rho}^u$ is sufficiently large to prevent overbooking, i.e. $\bar{\rho}^u > \max_{i=1,\dots,i_{\max}} u_i(\rho_1)$. Each policy π^{u*} formed by actions $a^{u*} = f^{u*}_i(c,d)$ each maximizing the right hand side of (3) leads to V^{u*}. We will call such a policy π^{u*} utility-optimal.

2.2 Structural Results

V^{u*} can be proven to be increasing and concave in c. In particular, the following theorem holds, see [1] for a proof.

Theorem 2. *Assume a decision-maker who maximizes expected additive time-separable utility with increasing and concave one-stage utility functions u_i. There then exists an utility-optimal policy $\pi^{u*} = (f^{u*}_N, f^{u*}_{N-1}, \dots, f^{u*}_1)$ for the static capacity control problem such that*

$$f_i^{u*}(c,d) = \begin{cases} \min\{d, c - y_{i-1}^{u*}(c)\} & c > y_{i-1}^{u*}(c) \\ 0 & c \leq y_{i-1}^{u*}(c) \end{cases},$$

with capacity dependent controls $y_{i-1}^{u*}(c) = \max\{y \in \{0,\ldots,(i-1)d_{\max}\} : u_i((c-y+1)\rho_i) - u_i((c-y)\rho_i) < \sum_{d'=0}^{d_{\max}} P(D_{i-1} = d')[V_{i-1}(y) - V_{i-1}(y-1)]\}$ *and* $y_0^{u*}(c) = 0$. *In addition, for all* i *and* c, $0 \leq y_{i-1}^{u*}(c+1) - y_{i-1}^{u*}(c) \leq 1$.

Note that given a decision-maker with additive time-separable utility function, there need not exist an utility-optimal policy that can be described in terms of (capacity-independent) protection levels for the static capacity control model. One can think of this as a consequence of the additive time-separable utility function that is composed of concave one-stage utility functions. The concavity of u induces intertemporal preferences and risk-aversion at the same time. The concave utility functions impose a preference for a smooth income stream over time and destroy the structure known from the expected revenue maximizing setting.

2.3 A Numerical Example

Given a total capacity of $C = 50$, seats are sold in $i_{\max} = 4$ booking classes at fares $\rho_1 = 1000$, $\rho_2 = 101$, $\rho_3 = 100$, and $\rho_4 = 10$. The total demand at each stage $i = 1,\ldots,4$ is assumed to be i.i.d. with $P(D_i = d) = 0.2$ for $d = 0,1,2$; $P(D_i = d) = 0.1$ for $d = 3,4$; $P(D_i = d) = 0.05$ for $d = 5$; and $P(D_i = d) = 0.01$ for $d = 6,\ldots,20$. The optimal risk-neutral protection levels are $y_1^* = y_2^* = 0$ and $y_3^* = 18$. In line with the results of Theorem 1, these controls are independent of c and increasing in the booking class i.

For an expected utility maximizing decision-maker with exponential one-stage utility function $u_i(w) = 1 - \exp(-0.05w)$ for all $i = 0,\ldots,4$, controls of $y_1^{u*}(c) = 0$ for all c are still preferred. The number of seats that should be protected for classes 2 and 3, however, depends on the number of seats available at the corresponding stage. Figure 1 shows a plot of the utility-optimal controls $y_2^{u*}(c)$ and $y_3^{u*}(c)$. The controls of booking class 2 are plotted in gray, the controls of class 3 are white, and dotted columns indicate that both are equal. Although it is obvious that the add-optimal controls depend on c, it can be seen that they never increase by more than 1. In addition, they need not be monotone in the booking class. In this example, $y_2^{u*}(c)$ is larger than $y_3^{u*}(c)$ for small values of c. They are equal for values of $38 \leq c \leq 41$. For higher values of c, the control $y_3^{u*}(c)$ is larger than $y_2^{u*}(c)$.

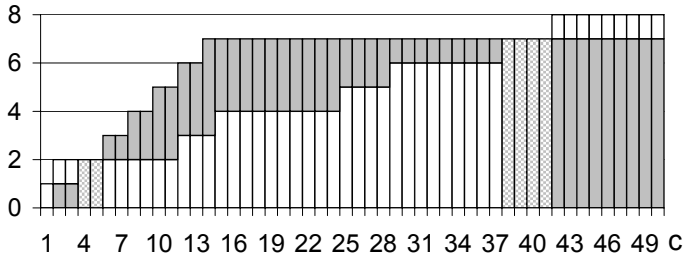


Fig. 1. Capacity dependent controls for classes 2 and 3 of an utility-optimal policy given exponential one-stage utility functions with $\gamma = 0.05$

3 Conclusion

Many applications might challenge the assumptions underlying expected additive time-separable utility maximization in a risk-averse context (despite of its analytical tractability).

In the static model, the periods might be so short that temporal preferences seem unlikely. Sometimes, different booking classes might even be open at the same time, and the protection levels determined by the static model are used simultaneously as a worst-case heuristic (because the order of arrival is the least preferred). Additive time-separable utility functions are inappropriate in these cases.

In this spirit, one could think of a decision-maker with a utility function that evaluates the total revenue gained independent of the timing within the booking horizon. Capacity control from the perspective of such a decision-maker is discussed e.g. in [1] and [2].

References

1. Barz C (2007) Risk-averse capacity control in revenue management. Lecture notes in economics and mathematical systems, Vol. 597, Springer, Berlin Heidelberg New York
2. Barz C, Waldmann K-H (2007) Risk-sensitive capacity control in revenue management. Mathematical Methods of Operations Research 65: 565–579
3. Gollier C (2001) The economics of risk and time. MIT Press, Cambridge
4. Rust J (1996) Numerical dynamic programming in economics. In: Amman, HM, Kendrick, DA, Rust, J (eds) Handbook of computational Economics. Elsevier, Amsterdam: 619–729
5. Talluri KT, van Ryzin GJ (2004) The theory and practice of revenue management. Kluwer, Boston
6. Weatherford LR (2004) EMSR versus EMSU: revenue or utility? Journal of Revenue and Pricing Management 3: 277–284

Routing and Capacity Optimization for IP Networks

Andreas Bley

Konrad-Zuse-Zentrum für Informationstechnik, Takustr. 7, 14195 Berlin, Germany. bley@zib.de

The world-wide Internet is a huge, virtual network comprised of more than 13,000 distinct networks, which all rely on the Internet Protocol (IP) for data transmission. Shortest path routing protocol such as OSPF or IS-IS control the traffic flow within most of these networks. The network administrator can manage the routing in these networks only by supplying a so-called *routing metric*, which specifies the link lengths (or routing weights) used in the shortest path computation.

The simplicity of this policy offers many operational advantages. ¿From the network planning perspective, however, shortest path routing is extremely complicated. As all routing paths depend on the same shortest path metric, it is not possible to configure the end-to-end routing paths for different communication demands individually. The routing can be controlled only indirectly and only as a whole by modifying the routing metric. Additional difficulties arise if each traffic demand must be sent unsplit via a single path – a requirement that is often imposed in practice to simplify network management and to avoid out-of-order packets and other undesired effects of traffic splitting. In this routing variant, the metric must be chosen such that all shortest paths are uniquely determined.

In this paper, we describe the main concepts and techniques that have been developed in [5] to solve dimensioning and routing optimization problems for such networks. We first discuss some fundamental properties of shortest path routings and the computational complexity of some basic network planning problems for this routing type. Then we describe an integer-linear programming approach to solve such problems in practice, which has been used successfully in the planning of the German national education and research network for several years.

1 Metrics and Routing Paths

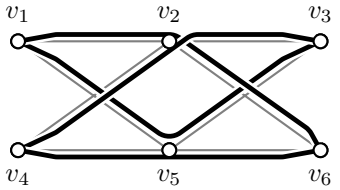
Given a digraph $D = (V, A)$ and a set K of directed commodities, an *unsplittable shortest path routing (USPR)* is a set of flow paths $P^*_{(s,t)}$, $(s,t) \in K$, such that there exists a *compatible metric* $\boldsymbol{\lambda} = (\lambda_a) \in \mathbb{Z}^A_+$ with respect to which each $P^*_{(s,t)}$ is the unique shortest (s,t)-path. One of the elementary problems in planning shortest path networks is to decide whether a given path set $\mathcal{S}$ is an USPR and, if so, to find a *compatible* routing metric $\boldsymbol{\lambda}$.

If there is no upper bound on the length values λ_a, this so-called Inverse Unique Shortest Paths problem can be solved very efficiently with linear programming techniques. We denote with $s(P)$ and $t(P)$ the start and end node of a path P, respectively, and with $\mathcal{P}(s,t)$ the set of all (s,t)-paths in D. It is not difficult to see that there exist an integer-valued metric compatible with $\mathcal{S}$ if and only if the following linear program has a solution [1]:

$$\begin{aligned} &\min \lambda_{\max} \\ \sum_{a \in P'} \lambda_a - \sum_{a \in P} \lambda_a &\geq 1 \qquad \forall\ P \in \mathcal{S},\ P' \in \mathcal{P}\big(s(P), t(P)\big) \setminus \{P\} \qquad (1) \\ 1 \leq \lambda_a &\leq \lambda_{\max} \quad \forall\ a \in A, \end{aligned}$$

Although this linear program contains exponentially many inequalities of type (1) it can be solved (or proven infeasible) in polynomial time; the separation problem for these inequalities reduces to $|\mathcal{S}|$ many 2-shortest path computations. Its (possibly fractional) optimal solution $\boldsymbol{\lambda}^*$ easily can be turned into an integer-valued, compatible metric by multiplying all values λ_a with a sufficiently large number and then rounding them to the nearest integer. As shown in [1], this approach yields a metric whose lengths exceed the lengths of the smallest possible integer-valued metric by a factor of at most $\min\left(|V|/2,\ \max\{|P| : P \in \mathcal{S}\}\right)$. For real-world problems, the lengths obtained this way are small enough to easily fit into the data formats of current routing protocols. In theory, however, the problem of finding a compatible routing metric with integer lengths as small as possible or bounded by a given constant is $\mathcal{NP}$-hard [5, 4].

If the given path set $\mathcal{S}$ is no unsplittable shortest path routing, then the above linear program is infeasible. Using standard greedy techniques, one then can construct from the final dual solution a subset $\mathcal{R}$ of the given paths, such that the paths in $\mathcal{R}$ cannot occur together in any USPR, but any proper subset of the path in $\mathcal{R}$ can. Figure 1 shows such an inclusion-wise minimal conflict set $\mathcal{R}$ consisting of four paths.



$P_1 = (v_1, v_2, v_6)$
$P_2 = (v_1, v_5, v_3)$
$P_3 = (v_4, v_2, v_3)$
$P_4 = (v_4, v_5, v_6)$

Fig. 1. Four paths that cannot occur together in a unique shortest path routing, but any subset of at most three of these paths can

These minimal conflict sets are of great practical importance. For every given digraph $D = (V, A)$, the family of all path sets that comprise a valid USPR forms an independence system (or hereditary family), and the circuits of this independence system are exactly these minimal conflict set. Any path set $\mathcal{S}$ that is not an USPR contains at least one of these minimal conflict sets. In a routing optimization framework, it hence is sufficient to ensure that none of these elementary conflicts occurs in the set of chosen routing paths to guarantee the these paths indeed from a valid USPR.

Several types of such elementary conflicts have been studied in the literature. The simplest one is a violation of the so-called *Bellman-* or *subpath-condition* [1, 8]: Two paths P_1 and P_2 can occur together in an USPR only if their (u, v)-subpaths $P_1[u, v]$ and $P_2[u, v]$ – if existent – coincide for all node pairs u, v. All elementary conflicts that involve only two paths are violations of the Bellman-condition. Generalizations of this condition are discussed in [1, 5], another type of necessary conditions has been studied in [10].

However, none of these combinatorial conditions yields a complete combinatorial description of all unsplittable shortest paths routings in a given digraph. In general, the minimal conflict sets can be very complex and arbitrarily large. Given an arbitrary path set $\mathcal{S}$, it is $\mathcal{NP}$-hard to approximate the size $|\mathcal{R}|$ of the smallest conflict set $\mathcal{R} \subseteq \mathcal{S}$ within a factor less than 7/6. The contrary problem of finding the largest subset $\mathcal{R} \subseteq \mathcal{S}$ that still comprises an USPR is computational hard as well. This problem cannot be approximated within a factor less than 8/7, unless $\mathcal{P} = \mathcal{NP}$ [5].

2 Hardness and Approximability

Network design and routing optimization problems with unsplittable shortest path routing are very difficult – from both the theoretical

and the practical point of view. In [5] three basic problem versions are thoroughly analyzed.

In the congestion minimization problem Min-Con-USPR, we are given a digraph $D = (V, A)$ with fixed arc capacities u_a and a set K of directed commodities with demand values d_{st}, and we seek for an USPR that minimizes the peak congestion (i.e., the maximum flow to capacity ratio over all arcs). This problem corresponds to the task of finding an efficient USPR in an existing network. The peak congestion is a good measure for the service quality network. In general, this problem is $\mathcal{NP}$-hard to approximate within a factor of $\mathcal{O}(|V|^{1-\epsilon})$ for any $\epsilon > 0$, but polynomially approximable within $\min(|A|, |K|)$.

Two extremal versions of designing and dimensioning an USPR network are expressed as the fixed charge network design problem FC-USPR and as the capacitated network design problem Cap-USPR, respectively. In both problems we are given a digraph with arc capacities and arc costs and a set of directed commodities with demand values. In FC-USPR the capacities are fix, and the goal is to find a minimum cost subgraph that admits an USPR of the commodities. This problem is $\mathcal{NPO}$-complete even if the underlying graph is an undirected ring or a bidirected cycle. In the capacitated network design problem Cap-USPR, we consider the given arc capacities as basic capacity units and seek a minimum cost installation of integer multiples of these basic capacity units, such that the resulting capacities admit an USPR of the given commodities. This problem cannot be approximated within a factor of $\mathcal{O}(2^{log^{1-\epsilon}|V|})$ in the directed and within a factor of $2-\epsilon$ in the undirected case, unless $\mathcal{P} = \mathcal{NP}$. For various special cases, however, better approximation algorithms can be derived. For the case where the underlying network is an undirected cycle or a bidirected ring, for example, Min-Con-USPR and Cap-USPR are approximable within constant factors.

The very restricted possibilities to configure the routing make unsplittable shortest path routing problems not only theoretically very hard, they are also an inherent drawback compared to other routing schemes in practice. In certain cases, these restrictions necessarily lead to unbalanced traffic flows with some highly congested links. In [3], we present a class of examples where the minimum congestion that can be obtained with unsplittable shortest path routing exceeds the congestion achievable with multicommodity flow, unsplittable flow, or shortest multi-path routing by a factor of $\Omega(|V|^2)$.

3 Solution Approaches

Traditional planning approaches for shortest path networks use local search, simulated annealing, or Lagrangian relaxation techniques with the lengths of the routing metric as primary decision variables [2, 6, 11, 12, 13, 15]. Basically, these approaches generate or iteratively modify numerous routing metrics and evaluate the resulting routings. The search for promising metrics is guided by the subgradients observed at the solution or other simple, local improvement criteria. The main challenges are to speed up the evaluation of the generated solution candidates to avoid the creation of poor candidates. The major drawbacks of these approaches are that they deliver no or only very weak quality guarantees for the computed solutions and that they perform well only for "easy" problems, where a globally efficient routing metric actually can be found by iterating simple local improvements.

In order to compute provenly optimal solutions, we propose a solution approach that – similar to Bender's decomposition – decomposes the routing subproblem into the two tasks of first finding the optimal end-to-end routing paths and then, secondly, finding a compatible routing metric for these paths.

In the master problem, we consider only the decisions concerning the design and dimensioning of the network and the choice of end-to-end routing paths. This part is solved using combinatorial methods and advanced integer linear programming techniques, which finally guarantees the optimality of the solution by this approach.

The client problem consists in finding a compatible routing metric for the end-to-end paths computed in the master problem. Whenever during the solution of the master problem an integer routing is constructed, we solve the client problem to determine whether the corresponding path set is a valid routing or not. This is done using the linear programming techniques illustrated in Section 1. If the current path set is an USPR, then we have found a incumbent solution for the master problem and the client problem's solution yields a compatible metric. Otherwise the client problem yields a minimal conflict among the current paths, which leads to an inequality that is valid for all USPRs, but violated by the current routing. Adding this inequality to the master problem, we cut off the current invalid solution and re-optimize the master problem. This approach was first described in [8] and refined and adapted to similar routing problems in [5, 14, 16].

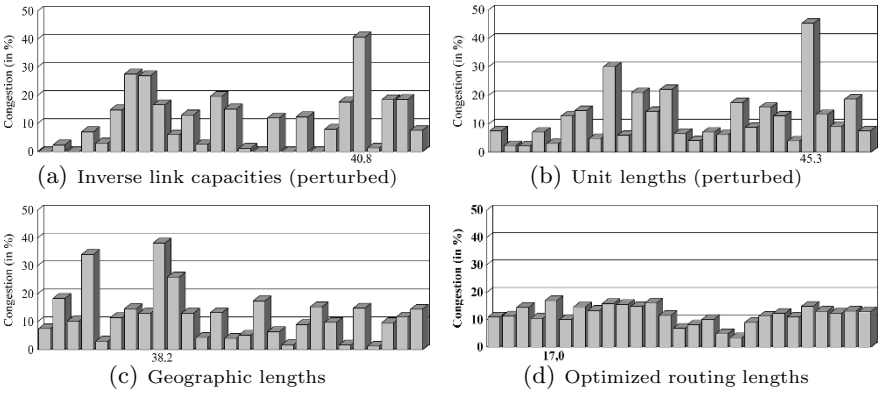
To illustrate this approach, consider the Min-Con-USPR problem introduced in the previous section. With $\mathcal{C}$ denoting the family of all (inclusion-wise) minimal path sets that cannot occur in an USPR, this

problem can be formulated as in integer programming problem as follows:

$$
\begin{aligned}
&\min L \\
&\sum_{P \in \mathcal{P}(s,t)} x_P = 1 && \forall\, (s,t) \in K \\
\sum_{(s,t)\in K} \sum_{P \in \mathcal{P}(s,t):\, a \in P} & d_{st}\, x_P \le u_a L && \forall\, a \in A \\
&\sum_{P \in \mathcal{S}} x_P \le |\mathcal{S}| - 1 && \forall\, \mathcal{S} \in \mathcal{C} \qquad (2) \\
&x_P \in \{0,1\} && \forall\, P \in \bigcup_{(s,t)\in K} \mathcal{P}(s,t) \\
&L \ge 0
\end{aligned}
$$

In principle, our decomposition approach solves this model with a branch-and-bound approach that dynamically separates violated conflict constraints (2) via the client problem. The initial formulation of the master problem would contain only the path variables for each commodity and some of the conflict inequalities (2), for example those corresponding to the Bellman-condition. At each node of the branch-and-bound tree we solve the current LP relaxation, pricing in path variables as needed. Whenever an integer solution x is found, we solve the linear program for the corresponding Inverse Unique Shortest Paths to find a compatible metric for the corresponding routing. If there exists one, then x yields the new incumbent solution for the master problem. If there is no compatible metric, we generate a violated conflict inequality (2) from the dual solution of the Inverse Unique Shortest Paths LP, add this inequality to the formulation of the master problem, and proceed with the branch-and-bound algorithm.

From the theoretical point of view, this approach seems not very attractive. For the plain integer programming formulation illustrated above, the integrality gap of the master problem can be arbitrarily large, the separation problem for the conflict inequalities is $\mathcal{NP}$-hard, and the optimal bases of the linear relaxation may necessarily become exponentially large. Nevertheless, carefully implemented this approach works surprisingly well for real-world problems. Our software implementation uses alternatively either a formulation based on path-routing variables or a formulation based on arc-routing variables for the master problem. In the branch-and-cut or branch-and-price-and-cut algorithms, we use specially tailored branching and pricing schemes as well as additional problem-specific primal heuristics and strong cutting planes. One type of these cutting planes, for example, exploits the spe-



(a) Inverse link capacities (perturbed) (b) Unit lengths (perturbed)

(c) Geographic lengths (d) Optimized routing lengths

Fig. 2. Link congestion values in G-WiN for several routing metrics

cial structure of the precedence constrained knapsacks defined by a link capacity constraint and the Bellman-condition among the paths across that link. In order to handle real-world problems, we also incorporated a very detailed and flexible hardware model, network failure resilience conditions, and various other types of technical and operational constraints into our software.

Numerous small and medium size benchmark problems could be solved optimally with this implementation. Even for large problems, for which optimality was not always achieved, our approach found better solutions than traditional metric-based methods in reasonable computation times. For several years, this software implementation has been used in the planning of the German national education and research networks B-WiN, G-WiN and X-WiN [6, 7, 9].

Figure 2 illustrates the importance of optimizing the routing in practice. It shows the different link loads that would result from the three most commonly used default settings for the routing metric and those resulting from the optimal routing metric for the G-WiN network with capacities and traffic demands of August 2001. The traffic is distributed much more evenly for the optimized metric. The peak congestion is not even half of that for the default settings, which significantly reduces packet delays and loss rates and improves the network's robustness against unforeseen traffic changes and failures.

References

1. W. Ben-Ameur and E. Gourdin. Internet routing and related topology issues. *SIAM Journal on Discrete Mathematics*, 17:18–49, 2003.

2. A. Bley. A Lagrangian approach for integrated network design and routing in IP networks. In *Proceedings of the 1st International Network Optimization Conference (INOC 2003), Paris, France*, pages 107–113, 2003.
3. A. Bley. On the approximability of the minimum congestion unsplittable shortest path routing problem. In *Proceedings of the 11th Conference on Integer Programming and Combinatorial Optimization (IPCO 2005), Berlin, Germany*, pages 97–110, 2005.
4. A. Bley. Inapproximability results for the inverse shortest paths problem with integer lengths and unique shortest paths. *Networks*, 50:29–36, 2007.
5. A. Bley. *Routing and Capacity Optimization for IP Networks.* PhD thesis, Technische Universität Berlin, 2007.
6. A. Bley, M. Grötschel, and R. Wessäly. Design of broadband virtual private networks: Model and heuristics for the B-WiN. In *Robust Communication Networks: Interconnection and Survivability*, volume 53 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 1–16. American Mathematical Society, 1998.
7. A. Bley and T. Koch. Optimierung des G-WiN. *DFN-Mitteilungen*, 54:13–15, 2000.
8. A. Bley and T. Koch. Integer programming approaches to access and backbone IP-network planning. ZIB Preprint ZR-02-41, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 2002.
9. A. Bley and M. Pattloch. Modellierung und Optimierung der X-WiN Plattform. *DFN-Mitteilungen*, 67:4–7, 2005.
10. P. Broström and K. Holmberg. Stronger necessary conditions for the existence of a compatible OSPF metric. Technical Report LiTH-MAT-R-2004-08, Linköping University, May 2004.
11. L. Buriol, M. Resende, C. Ribeiro, and M. Thorup. A hybrid genetic algorithm for the weight setting problem in OSPF/IS-IS routing. *Networks*, 46:36–56, 2005.
12. M. Ericsson, M. Resende, and P. Pardalos. A genetic algorithm for the weight setting problem in OSPF routing. *Journal of Combinatorial Optimization*, 6:299–333, 2002.
13. B. Fortz and M. Thorup. Increasing Internet capacity using local search. *Computational Optimization and Applications*, 29:13–48, 2004.
14. K. Holmberg and D. Yuan. Optimization of Internet protocol network design and routing. *Networks*, 43:39–53, 2004.
15. F. Lin and J. Wang. Minimax open shortest path first routing algorithms in networks suporting the SMDS service. In *Proceedings of the IEEE International Conference on Communications 1993 (ICC'93), Geneva, Suisse*, volume 2, pages 666–670, 1993.
16. M. Prytz. *On Optimization in Design of Telecommunications Networks with Multicast and Unicast Traffic.* PhD thesis, Royal Institute of Technology, Stockholm, Sweden, 2002.

Coping with Incomplete Information in Scheduling – Stochastic and Online Models*

Nicole Megow

Institut für Mathematik, Technische Universität Berlin, Germany
nicole.megow@tu-berlin.de

Incomplete information is an omnipresent issue when dealing with real-world optimization problems. Typically, such limitations concern the uncertainty of given data or the complete lack of knowledge about future parts of a problem instance. Our work is devoted to investigations on how to cope with incomplete information when solving scheduling problems. The particular problem class we consider is the class of machine scheduling problems which plays an important role within combinatorial optimization. These problems involve the temporal allocation of limited resources (machines) for executing activities so as to optimize some objective. Scheduling problems are apparent in many applications including, for example, manufacturing and service industries but also compiler optimization and parallel computing.

There are two major frameworks for modeling limited information in the theory of optimization. One deals with *stochastic information*, the other with *online information*. Within these models, we design algorithms for certain scheduling problems. Thereby we provide first constant performance guarantees or improve previously best known results.

Both frameworks have their legitimacy depending on the actual application. Nevertheless, problem settings are conceivable that comprise both, uncertain information about the data set and the complete lack of knowledge about the future. This rouses the need for a generalized model that integrates both traditional information environments. Such a general model is designed as a natural extension that combines stochastic and online information. The challenging question is whether there exists any algorithm that can perform well in such a restricted

* This work has been supported by the DFG Research Center Matheon in Berlin. The book version of this extended abstract is published as [6].

information environment. More precisely, is there an algorithm that yields a constant performance guarantee? We successfully treat this intriguing question and give a positive answer by providing such algorithms for certain machine scheduling problems. In fact, our results are competitive with the performance guarantees best known in the traditional settings of stochastic and online scheduling. Thus, they do not only justify the generalized model but also imply – at least in the considered problem settings – that optimization in the general model with incomplete information does not necessarily mean to give up performance.

1 Stochastic Scheduling

In stochastic scheduling we assume uncertainty about job processing times. Any job j must be processed for P_j units of time, where P_j is a random variable. We assume that all random variables of processing times are stochastically independent. This restriction on the probability functions is not part of the stochastic scheduling model; still, the independence of random variables is crucial for our and previously known results.

The solution of a stochastic scheduling problem is not a simple schedule, but a so-called *scheduling policy*; see [10]. A policy must not anticipate information about the future, such as the actual realizations of the processing times of the jobs that have not yet been completed; we say a stochastic scheduling policy must be *non-anticipatory*.

Various research on stochastic scheduling has been published concerning criteria that guarantee the optimality of simple policies for rather special, restricted scheduling problems. Only recently research interest addressed also approximative policies [11, 13, 2]. While all of the results hold for non-preemptive scheduling, we are not aware of any approximation results for problems that allow job preemption except from the optimality of the Gittins index priority policy [12, 15, 4] for the problem $1 \,|\, \mathrm{pmtn} \,|\, \mathbb{E}\,[\sum w_j C_j\,]$.

We derive first constant approximation guarantees for preemptive stochastic scheduling policies on multiple machines and/or individual release dates. For jobs with general processing time distributions, we give a 2-approximative policy for minimizing the expected sum of weighted completion times.

In order to derive our results we introduce a new non-trivial lower bound on the expected value of an unknown optimal policy. This bound is obtained borrowing ideas for a *fast single-machine relaxation* [1]. The

crucial ingredient to our investigations is then the application of the above mentioned Gittins index priority policy which solves a relaxed version of our fast single-machine relaxation optimally [12, 15, 4]. The priority index used in this policy also inspires the design of our policies. Thereby, our preemptive policies extensively utilize information on processing time distributions other than the first (and second) moments, which distinguishes them significantly from approximative policies known in the non-preemptive setting.

The Gittins index is defined as follows. Given that a job j has been processed for y time units and it has not completed, we define the *expected investment* of processing this job for q time units or up to completion, which ever comes first, as

$$I_j(q, y) = \mathbb{E}\left[\min\{\mathrm{P}_j - y, q\} \,|\, \mathrm{P}_j > y\right].$$

The ratio of the weighted probability that this job is completed within the next q time units over the expected investment, is the basis of the Gittins index priority rule. We define it as the *rank* of a sub-job of length q of job j, after it has completed y units of processing:

$$R_j(q, y) = \frac{w_j \Pr\left[\mathrm{P}_j - y \leq q \,|\, \mathrm{P}_j > y\right]}{I_j(q, y)}.$$

This ratio is well defined if we assume that we compute the rank only for $q > 0$ and $\mathrm{P}_j > y$, in which case the investment $I_j(q, y)$ has a value greater than zero.

For a given (unfinished) job j and attained processing time y, we are interested in the maximal rank it can achieve. We call this the Gittins index, or rank, of job j, after it has been processed for y time units.

$$R_j(y) = \max_{q \in \mathbb{R}^+} R_j(q, y).$$

With the definitions above, we define a policy based on the rank for scheduling on parallel machines where jobs have release dates.

Follow Gittins Index Priority Policy (F-Gipp): At any time t, process an available job j with highest rank $R_j(y_{j,k+1})$, where (j, k) is the last quantum of j that has completed and $y_{j,k+1}$ is the amount of processing that has been completed before the next quantum $(j, k+1)$ starts. Define $k = 0$ if no quantum of job j has been completed.

The policy F-Gipp is a 2-approximation for the preemptive stochastic scheduling problem $\mathrm{P} \,|\, r_j, pmtn \,|\, \mathbb{E}\left[\sum w_j C_j\right]$. However, on restricted problem instances it coincides with policies whose optimality is known; see [9, 6].

2 Online Scheduling

In online scheduling we assume that jobs and their characterizing data become known to the scheduler only piecewise. Thus, an online algorithm must take scheduling decisions based only on the partial knowledge of the instance as it is given so far.

We investigate algorithms for scheduling with the objective to minimize the total weighted completion time on single as well as on parallel machines. We consider both, a setting with independent jobs and one where jobs must obey precedence relations.

For independent jobs arriving online, we design and analyze algorithms for both, the preemptive and the non-preemptive setting. These online algorithms are extensions of the classical Smith rule [14] and yield performance guarantees that are improving on the previously best known ones. A natural extension of Smith's rule to the preemptive setting is 2-competitive. For the non-preemptive variant of the multiple-machine scheduling problem, we derive a 3.281-competitive algorithm that combines a processing time dependent waiting strategy with Smith's rule.

We are not aware of any existing results for the scenario in which precedence constraints among jobs are given. We discuss a reasonable online model and give lower and upper bounds on the competitive ratio for scheduling without job preemptions. In this context, previous work on the offline problem of scheduling jobs with generalized precedence constraints, the so called AND/OR-precedence relations [3], appears to be adoptable to a certain extent.

3 Stochastic Online Scheduling

We consider the *stochastic online scheduling* (SOS) model that generalizes both traditional models for dealing with incomplete information, stochastic scheduling and online scheduling. Like in online scheduling, we assume that the instance is presented to the scheduler piecewise, and nothing is known about jobs that might arrive in the future. Even the number of jobs is not known in advance. Once a job arrives, we assume, like in stochastic scheduling, that the probability distribution of its processing time is disclosed, but the actual processing time remains unknown until the job completes.

The goal is to find an SOS policy that minimizes the expected objective value. Our definition of a stochastic online scheduling policy integrates the traditional definition of stochastic scheduling policies into

the setting where jobs arrive online. In order to decide, such a policy may utilize the complete information contained in the partial schedule up to time t. But it must not utilize any information about jobs that will be released in the future and it must not use the actual processing times of scheduled (or unscheduled) jobs that have not yet completed. In the performance evaluation we also generalize the definitions of an approximative policy for stochastic scheduling and a competitive algorithm in online scheduling; see [8, 6]. In this view, our model somewhat compares to the idea of a *diffuse adversary* as defined by Koutsoupias and Papadimitriou [5].

Various (scheduling) problems can be modeled in this stochastic online setting. We consider the particular settings of preemptive and non-preemptive scheduling with the objective to minimize the expected total weighted completion times of jobs.

For the problem where jobs must run until completion without interruption, $\mathrm{P} \,|\, r_j \,|\, \mathbb{E}\,[\sum w_j C_j\,]$, we analyze simple, combinatorial online scheduling policies and derive performance guarantees that match the currently best known performance guarantees for stochastic and online parallel-machine scheduling. For processing times that follow NBUE distributions, a MinIncrease policy even improves upon previously best known performance bounds from stochastic scheduling, even though it is feasible in a more general setting. This policy assigns each job j to the machine where it causes the least increase in the expected objective value, given the previously assigned jobs (when release dates are ignored). In the analysis we exploit the fact that the lower bound for an optimal policy in the traditional stochastic scheduling environment in [11] is by definition also a lower bound for an optimal policy in the Sos model.

In the preemptive setting we can argue that the 2-approximative policy for preemptive stochastic (offline) scheduling in Section 1 for $\mathrm{P} \,|\, r_j, \mathrm{pmtn} \,|\, \mathbb{E}\,[\sum w_j C_j\,]$ also applies in this more general model because the preemptive policy is feasible in an online setting as well. Moreover, the currently best known online algorithm for deterministic processing time has also a competitive ratio of 2; see [7].

4 Conclusion

These results do not only justify the general model for scheduling with incomplete information. They also show for certain scheduling problems that policies designed to deal with stochastic *and* online information, can achieve the same theoretic performance guarantee as policies that can handle only one type of limited knowledge.

References

1. C. Chekuri, R. Motwani, B. Natarajan, and C. Stein. Approximation techniques for average completion time scheduling. *SIAM Journal on Computing*, 31:146–166, 2001.
2. B. C. Dean. *Approximation Algorithms for Stochastic Scheduling Problems*. PhD thesis, Massachusetts Institute of Technology, 2005.
3. T. Erlebach, V. Kääb, and R. H. Möhring. Scheduling AND/OR-networks on identical parallel machines. In *Proc. of the First International Workshop on Approximation and Online Algorithms, WAOA 2003*, volume 2909 of *Lecture Notes in Computer Science*, pages 123–136, 2004.
4. J. C. Gittins. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society*, Series B, 41:148–177, 1979.
5. E. Koutsoupias and C. H. Papadimitriou. Beyond competitive analysis. *SIAM Journal on Computing*, 30:300–317, 2000.
6. N. Megow. *Coping with Incomplete Information in Scheduling – Stochastic and Online Models*. Dissertation 2006, Technische Universität Berlin. Cuvillier Göttingen, 2007.
7. N. Megow and A. S. Schulz. On-line scheduling to minimize average completion time revisited. *Operations Research Letters*, 32(5):485–490, 2004.
8. N. Megow, M. Uetz, and T. Vredeveld. Models and algorithms for stochastic online scheduling. *Mathematics of Operations Research*, 31(3):513–525, 2006.
9. N. Megow and T. Vredeveld. Approximation in preemptive stochastic online scheduling. In *Proc. of 14th European Symposium on Algorithms*, number 4168 in Lecture Notes in Computer Science, pages 516–527, 2006.
10. R. H. Möhring, F. J. Radermacher, and G. Weiss. Stochastic scheduling problems I: General strategies. *Zeitschrift für Operations Research*, 28:193–260, 1984.
11. R. H. Möhring, A. S. Schulz, and M. Uetz. Approximation in stochastic scheduling: the power of LP-based priority policies. *Journal of the ACM*, 46:924–942, 1999.
12. K. C. Sevcik. Scheduling for minimum total loss using service time distributions. *Journal of the ACM*, 21:65–75, 1974.
13. M. Skutella and M. Uetz. Stochastic machine scheduling with precedence constraints. *SIAM Journal on Computing*, 34(4):788–802, 2005.
14. W. E. Smith. Various optimizers for single-stage production. *Naval Research Logistics Quarterly*, 3:59–66, 1956.
15. G. Weiss. On almost optimal priority rules for preemptive scheduling of stochastic jobs on parallel machines. *Advances in Applied Probability*, 27:827–845, 1995.

Availability and Performance Analysis of Stochastic Networks with Unreliable Nodes

Cornelia Wichelhaus née Sauer

Institute of Applied Mathematics, University of Heidelberg, Im Neuenheimer Feld 294, D - 69120 Heidelberg. wichelhaus@statlab.uni-heidelberg.de

Summary. The article presents results of the PhD thesis [3] written at the University of Hamburg. It is devoted to the study of stochastic networks with unreliable servers at the nodes. It is shown that there exist large classes of degradable networks for which the steady state distribution is of product form leading to a comprehensive performance and availability analysis of the systems.

1 Introduction

Stochastic networks are systems of nodes between which customers move in order to receive service. Typical application fields are computer and telecommunication networks, the internet, manufacturing networks, logistics and supply chain networks, as well as population dynamical systems and social migration systems. For all such systems reliability and availability play a major role: Networks which we observe in reality are never totally reliable; there are interruptions of service or breakdowns of components of the system due to human or technical failures or external catastrophes. Then the performance of the system is degraded and its regular time-behavior is perturbed until the failure is repaired. Thus, while modeling such systems, right from the start reliability aspects should be taken into account in parallel to the classical paradigm of performance analysis. However, the factor reliability is not considered in the classical theory of stochastic networks and there are until now no striking results for systems with unreliable components. Exact modeling approaches of such systems for a unified investigation of performance and reliability aspects resisted up to now the derivation of easy to apply recipes for systems' analysis. On the other hand, explicit results for the (asymptotic) analysis are of great importance

since they open the way for a comprehensive understanding of the systems. To overcome the lack of explicit steady state results for unreliable models approximation techniques have been developed, see the survey [2].

The central contribution of the thesis [3] is to describe and analyze stochastic networks with unreliable nodes for which both, the performance behavior and the breakdown and repair mechanisms are incorporated into a unified Markovian system description, and which still show explicit access to the asymptotic and steady state behavior. For large classes of unreliable networks explicit steady state distributions are derived which are in parallel to the classical results of product form networks. Based on these explicit results, the conditions for the systems to stabilize as well as the interplay of performance evaluation and availability analysis can be studied. Moreover, in the combined performability models generalized concepts and ideas from the classical theory of product form networks can then be applied. These include the study of job-observer properties by means of Palm martingale calculus as well as the establishment of dependence ordering and comparison results for two degradable networks. We refer the reader to [3].

To illustrate the basic ideas of the approach we present here a prototype example of a degradable network of exponential type for which the steady state can be explicitly computed.

2 Degradable Exponential Networks of Product Form

Consider a network of J nodes summarized in the set $\bar{J} := \{1, 2, \ldots, J\}$. Station j is a single server with infinite waiting room under FCFS regime. Customers in the network are indistinguishable. At node j there is an external Poisson-λ_j-arrival stream, $\lambda_j \geq 0$. Customers arriving at node j from the outside or from inside of the network request for an amount of service time which is exponentially distributed with mean 1. Service at node j is provided with intensity $\mu_j(n_j) > 0$, if there are $n_j > 0$ customers present at node j, we set $\mu_j(0) := 0$. All service and interarrival times are assumed to be independent.

Movements of customers in the network are governed by a Markovian routing mechanism. A customer on leaving node i selects with probability $r(i, j) \geq 0$ to visit node j next, and then enters node j immediately, commencing service if he finds the server idle, otherwise he joins the tail of the queue at node j; with probability $r(i, 0) \geq 0$ the customer decides to leave the network immediately. Thus, for all $i \in \bar{J}$

we have $\sum_{j=0}^{J} r(i,j) = 1$. Given the departure node i the customer's routing decision is made independently of the network's history.

Let $\lambda := \sum_{j=1}^{J} \lambda_j$, $r(0,j) := \frac{\lambda_j}{\lambda}$ and $r(0,0) := 0$. We assume that the matrix $\mathcal{R} := (r(i,j) : i,j \in \bar{J} \cup \{0\})$ is irreducible and denote by $\eta = (\eta_1, \ldots, \eta_J)$ the unique solution of the traffic equation corresponding to the network in full availability status,

$$\eta_j = \lambda_j + \sum_{i=1}^{J} \eta_i \, r(i,j), \quad j \in \bar{J}.$$

Thus, the vector η gives the visiting rates of customers at the nodes in the network in full availability status.

The servers at the nodes are unreliable, i.e. the nodes may break down. The breakdown events are of rather general structure and may occur in different ways: Nodes may break down as an isolated event or in groups simultaneously and the repair of nodes may end for each node individually or in groups as well. It is not required that those nodes which stopped service simultaneously return to service at the same time instant. More precisely the **control of breakdowns and repairs** is:
Let $\bar{I} \subset \bar{J}$ be the set of nodes in down status with $n_i \in \mathbb{N}$ customers at node $i, i \in \bar{I}$, which are waiting there for service to be resumed.

- Let $\bar{K} \subset \bar{J} \backslash \bar{I}, \bar{K} \neq \emptyset$, be some subset of nodes in up status. Then the nodes of $\bar{K}$ break down concurrently with intensity $\alpha\left(\bar{I},\, \bar{I} \cup \bar{K},\, n_i : i \in \bar{J}\right)$, if there are n_i customers at node $i, i \in \bar{J}$.
- Let $\bar{H} \subset \bar{I}, \bar{H} \neq \emptyset$, be some subset of nodes in down status. Then the nodes of $\bar{H}$ return from repair as a batch group with intensity $\beta\left(\bar{I},\, \bar{I} \backslash \bar{H},\, n_i : i \in \bar{J}\right)$ and immediately resume their services.

Thus, breakdowns and repairs may depend on local loads of the corresponding nodes.

In this general setting the intensities $\alpha\left(\bar{I},\, \bar{I} \cup \bar{K},\, n_i : i \in \bar{J}\right)$ and $\beta\left(\bar{I},\, \bar{I} \backslash \bar{H},\, n_i : i \in \bar{J}\right)$ for occurrence of breakdown and repair events cannot be chosen arbitrarily, but have to meet some constraints. By considering breakdowns as births and repairs as deaths we have found versatile rules for rather general classes of suitable intensities by applying results from the theory of multi-dimensional birth-death processes. One possible definition:

Definition 1. *Assume that $\bar{I}, \bar{I} \subset \bar{J}$, is the set of nodes in down status. The intensities for breakdowns and repairs respectively are*

$$\alpha\left(\bar{I},\, \bar{I}\cup\bar{K},\, n_i : i\in\bar{J}\right) := \frac{A(\bar{I}\cup\bar{K},\, n_i : i\in\bar{I}\cup\bar{K})}{A(\bar{I},\, n_i : i\in\bar{I})} \quad \textit{and}$$

$$\beta\left(\bar{I},\, \bar{I}\backslash\bar{H},\, n_i : i\in\bar{J}\right) := \frac{B(\bar{I},\, n_i : i\in\bar{I})}{B(\bar{I}\backslash\bar{H},\, n_i : i\in\bar{I}\backslash\bar{H})} \quad \textit{respectively,}$$

where A and B are nonnegative functions, $A, B : \left(\mathcal{P}(\bar{J})\times\bigcup_{l\in\bar{J}_0}\mathbb{N}^l\right)\to\mathbb{R}_+=[0,\infty)$. We assume all intensities to be finite and define $A(\emptyset,\, n_i : i\in\emptyset) := 1 =: B(\emptyset,\, n_i : i\in\emptyset)$ and $\frac{0}{0} := 0$.

Nodes in down status neither accept new customers nor continue serving the old customers who have to wait there for the server's return. Thus, we have to reroute the customers in the network. We describe here three regimes to handle routing connected with nodes in down status. They are derived from principles used to resolve blocking situations in networks with resource constraints, see [1]. It is surprising that they can also be used here. Assume that $\bar{I}$ is the set of nodes in down status.

Definition 2 (Stalling). *If there is a breakdown of either a single node or a group of nodes, then all arrival streams to the network and all service processes at the nodes in up status are completely interrupted and resumed only after all failed nodes are repaired.*

Definition 3 (Blocking). *A customer after being served at node $i\in\bar{J}\backslash\bar{I}$ chooses the next destination node j according to the routing matrix $\mathcal{R}$. If node j is in down status, the customer stays at node i to obtain another service. When this additional service expires the customer selects his destination node anew according to $\mathcal{R}$.*

Definition 4 (Skipping). *If a customer at node $i\in\bar{J}\backslash\bar{I}$ selects for the next jump's destination node $j\in\bar{J}\backslash\bar{I}\cup\{0\}$ in up status the jump is allowed and immediately performed and the customer joins node j for service. If the customer selects for the next jump's destination node $k\in\bar{I}$, he only performs an imaginary jump to that node, spending no time at node k, but immediately performs the next jump according to the routing matrix $\mathcal{R}$, i.e. with probability $r(k,l)$ he selects the successor node l; if $l\in\bar{J}\backslash\bar{I}\cup\{0\}$ the jump is performed and the customer joins node l for service, but if $l\in\bar{I}$ the customer has to perform another random choice as if he would depart from node l; and so on.*

Remark 1. In [3], Chapter 2, we derive a characterization of the set of all rerouting mechanisms which are applicable here in case of a breakdown for controlling the movements of customers since they lead to explicit

access to the steady state of the corresponding unreliable network. We represent this set as a subset of an affine space in explicit terms which opens the way for optimization procedures when choosing the rerouting matrix. The rationale behind all suitable rerouting mechanisms is that the proportions with which customers are allocated to the respective nodes in up status remain unchanged and are in particular independent of the availability status of other nodes in the network. This property is in many cases desirable. Of course, the traffic intensities at the nodes may change in response to breakdowns or repairs.

For describing the system's evolution over the time axis we introduce states of the form

$$(\bar{I}, n_1, n_2, \ldots, n_J) \in \left(\mathcal{P}(\bar{J}) \times \mathbb{N}^J\right) =: \widetilde{E}.$$

The set $\bar{I}, \bar{I} \subset \bar{J}$, contains the nodes under repair. For node $j \in \bar{J} \backslash \bar{I}$ operating in a normal up status there are $n_j \in \mathbb{N}$ customers present and if $n_j > 0$ one of them is under service. For node $i \in \bar{I}$ in down status there are $n_i \in \mathbb{N}$ customers waiting for the return of the repaired server.

Operating on these states we define a Markov process $\widetilde{X} = (Y, X)$ describing the degradable network with state space $\widetilde{E}$ according to the rules just explained. The following theorem shows that modeling degradable networks in this way leads to explicit steady state distributions for the network process $\widetilde{X}$.

Theorem 1. *The describing network process $\widetilde{X}$ on the state space $\widetilde{E}$ has the stationary distribution*

$$\pi(\bar{I}, n_1, n_2, \ldots, n_J) \tag{1}$$
$$= C^{-1} \frac{A(\bar{I}, n_i : i \in \bar{I})}{B(\bar{I}, n_i : i \in \bar{I})} \prod_{j=1}^{J} \prod_{l=1}^{n_j} \left(\frac{\eta_j}{\mu_j(l)}\right) \text{ for } (\bar{I}, n_1, n_2, \ldots, n_J) \in \widetilde{E},$$

if and only if the normalization constant C is finite.

The stationary distribution is of a remarkable nice product form. It carries information about the performance and reliability behavior of the network and allows a combined performance and availability analysis of the system; availability measures and performance characteristics like average loads and throughputs can directly be computed.

Remark 2. Consider the special case of Def. 1 with breakdown and repair intensities which are independent of customer loads at the nodes,

$$A(\bar{K}, n_i : i \in \bar{K}) := A(\bar{K}) \text{ and } B(\bar{K}, n_i : i \in \bar{K}) := B(\bar{K}) \text{ for all } \bar{K} \subset \bar{J}.$$

Then the marginal breakdown and repair process Y is Markovian on its own on the state space $\mathcal{P}(\bar{J})$. It follows that in this case the steady state distribution (1) factorizes into the steady state distribution of Y and the steady state distribution of a classical exponential network with reliable nodes, but all performance parameters the same as for the network described by $\widetilde{X}$. This means that the degradable network is ergodic under the same conditions as the classical network. Furthermore, the steady state of the marginal queue length process X of the degradable network - which is not Markovian on its own - coincides with the equilibrium distribution of the classical network. Moreover, at a fixed time instant in equilibrium the processes X and Y behave as if they were independent which is not intuitive since the transition mechanisms of X strongly depend on the actual state of Y. As a consequence here the asymptotic performance and availability analysis can be decoupled.

3 Generalizations and Complements

The results of the previous section can be generalized to more versatile classes of network models which allow distinguishable classes of customers, general service time distributions and general service disciplines. Moreover, the up and down times of the nodes may obey general distributions or can be determined by generalized multi-dimensional migration or spatial processes. For details we refer the reader to [3].

Therewith we are able to model and analyze various classes of degradable complex systems with strongly dependent components which interact by moving customers as well as by common mode failures and simultaneous repairs.

References

1. Balsamo S, De Nitto Persone V, Onvural R (2001) Analysis of Queueing Networks with Blocking. Kluwer academic publishers
2. Chakka R, Mitrani I (1996) Approximate solutions for open networks with breakdowns and repairs. In: Kelly FP, Zachary S, Ziedins I (eds) Stochastic Networks - Theory and applications, chapter 16: 267-280, Clarendon Press, Oxford
3. Sauer C (2006) Stochastic product form networks with unreliable nodes: Analysis of performance and availability. PhD Thesis, Department of Mathematics, University of Hamburg

Part II

Diploma Award Winners

Heuristics of the Branch-Cut-and-Price-Framework SCIP

Timo Berthold

Konrad-Zuse-Zentrum für Informationstechnik Berlin, Germany
berthold@zib.de

Summary. In this paper we give an overview of the heuristics which are integrated into the open source branch-cut-and-price-framework SCIP. We briefly describe the fundamental ideas of different categories of heuristics and present some computational results which demonstrate the impact of heuristics on the overall solving process of SCIP.

1 Introduction

A lot of problems arising in various areas of Operations Research can be formulated as *Mixed Integer Programs (MIP)*. Although MIP-solving is an $\mathcal{NP}$-hard optimization problem, many practically relevant instances can be solved in reasonable time. The standard exact method for solving MIPs is *branch-and-cut*, a combination of LP-based branch-and-bound and cutting plane techniques. Besides that, heuristics (Greek $\epsilon\upsilon\rho\iota\sigma\kappa\epsilon\iota\nu$ – to find) are incomplete methods which quickly try to construct feasible solutions of high quality, but without any guarantee to find one.

In state-of-the-art MIP-solvers like the branch-cut-and-price-framework SCIP (Solving Constraint Integer Programs) [1, 3] heuristics play a major role in finding and improving feasible solutions at early stages of the solution process. This helps to reduce the overall computational effort, guides the remaining search process, and proves the feasibility of the MIP model. Furthermore, a heuristic solution with a small gap to optimality often is sufficient for practical applications.

Overall, there are 23 heuristics integrated into SCIP version 1.00. They can be roughly subclassified into four categories: rounding, diving, objective diving, and large neighborhood search heuristics. In the remainder, we will give a short introduction into these strategies and

afterwards we will present some computational results. For more detail, we refer to Achterberg [1] and Berthold [6].

2 Rounding Heuristics

All rounding heuristics in SCIP work in the following way: they take an LP-feasible but fractional point – normally the optimum of some LP-relaxation – and iteratively round the fractional variables. Thereby, the number of fractional variables is reduced one by one in each iteration step (except if a shift is performed, see below). Regarding rounding heuristics, the most important issue is, not to loose the LP-feasibility during the iteration process, or if so, try to immediately recover LP-feasibility.
There are four rounding heuristics in SCIP:

- *Simple Rounding* only performs roundings, which assure to keep feasibility;
- *Rounding* conducts roundings, which potentially violate some constraints and reduces existent violations by further roundings;
- *Shifting* is allowed to change (shift) the values of integral or continuous variables in order to recover feasibility;
- *Integer Shifting* proceeds like Shifting, but does not consider continuous variables. If it succeeds, it solves an LP in order to set the continuous variables to their optimal value.

Each of these procedures is an extension of the ones which are listed before it. The latter are more powerful, but also more expensive in terms of running time and therefrom they are applied less frequently.

3 Diving Heuristics

The principal idea of diving heuristics comes from the branch-and-bound procedure. They iteratively round some fractional variable and resolve the LP-relaxation, simulating a depth-first-search in the tree. In doing so, diving heuristics use a special branching rule which tends towards feasibility and not primary towards a good subdivision of the problem, as common branching rules do.

The six diving heuristics implemented in SCIP mainly differ in the applied branching rule. It chooses a variable with:

- *Fractional Diving:* smallest fractionality;
- *Coefficient Diving:* smallest number of potentially violated rows;

- *Linesearch Diving:* greatest difference of root solution and current LP solution;
- *Guided Diving:* smallest difference to the best known integral solution;
- *Pseudocost Diving:* smallest ratio of estimated objective increase if rounding to either direction;
- *Vectorlength Diving:* smallest ratio of potential objective change and number of affected constraints.

In [6], it is shown that none of them dominates the others in terms of performance.

4 Objective Diving Heuristics

Heuristics of this category iteratively manipulate the objective function and resolve the LP-relaxation in order to reach an integral vertex of the LP-polyhedron. They perform "soft roundings" by adding punishment terms to the objective instead of performing "hard roundings", i.e., fixing variables like the heuristics of Sections 2 and 3.

There are actually three objective diving heuristics in SCIP: Objective Pseudocost Diving, Rootsolution Diving and the Objective Feasibility Pump. In our computational studies, the latter one proved to be superior to the others.

The Feasibility Pump was first described by Fischetti et al. [10, 5], the version which is implemented in SCIP was introduced by Achterberg and Berthold [2]. By taking the original objective of the MIP into account, the Objective Feasibility Pump is able to produce solutions of a much better objective value in a comparable running time.

5 LNS Heuristics

Large neighborhood search (LNS) heuristics solve a sub-MIP of the original MIP in order to investigate a neighborhood of a special point, e.g., the best known integral solution (*incumbent*). This sub-MIP is created by fixing a sufficient number of variables or adding very restrictive constraints. The hope is that the sub-MIP is much easier to solve, but still contains solutions of high quality.

Four of the five LNS heuristics available in SCIP are improvement heuristics, i.e., they take some feasible solution as a starting point:

- *Local Branching* [11] adds a distance constraint which allows only a certain number of variables to differ from their value in the incumbent;
- *RINS* [9] fixes variables which take identic values in the current node's LP-relaxation and the incumbent;
- *Crossover* [6] fixes variables which take identic values in a certain number of feasible solutions;
- *Mutation* [6] randomly fixes variables to their incumbent value.

In contrast to these four, *RENS* [6, 7] is an LNS rounding heuristic. It fixes all variables which take integral values in the optimum of the LP-relaxation (often more than 90%) and changes the bounds to the nearest integers for fractional variables. This implies that all integer variables of the sub-MIP are binary.

By completely solving the RENS sub-MIP, one is able to determine whether a point can be rounded to an integral solution and which one is the best possible rounding. Furthermore, a slightly restricted version of RENS proves to be a reasonable start heuristic.

6 Computational Results

The computational experiments reported here were obtained with SCIP version 0.82b running on a 3.80 GHz Intel Pentium 4 with 2 GB RAM, using CPLEX 10.0 as underlying LP-solver. We chose a test set of 129 instances taken from the MIPLIB 3.0 [8], the MIPLIB2003 [4] and the MIP collection of Mittelmann [12].

First, we evaluated the individual impact of the 15 heuristics which are used by default. For each heuristic, we investigated the change of performance caused by deactivating it. We compared the geometric means of the running time and the number of branch-and-bound-nodes taken over the 97 instances which could be solved to optimality within an hour, using SCIP with default settings. For the other instances we compared the primal-dual gap after running SCIP for an hour.

We observed that deactivating a single heuristic only has a small impact; the geometric means of the running time and the number of branch-and-bound-nodes always changed by less than 5%, except for the Objective Feasibility Pump (12% and 30%, respectively).

On the other hand, deactivation of all available heuristics leads to a significant deterioration: the geometric mean of the running time and the number of branch-and-bound-nodes raises by a factor of two, the remaining gap by about 50%. There are considerably less instances

which are solved to optimality within an hour, or for which at least one feasible solution is found, respectively.

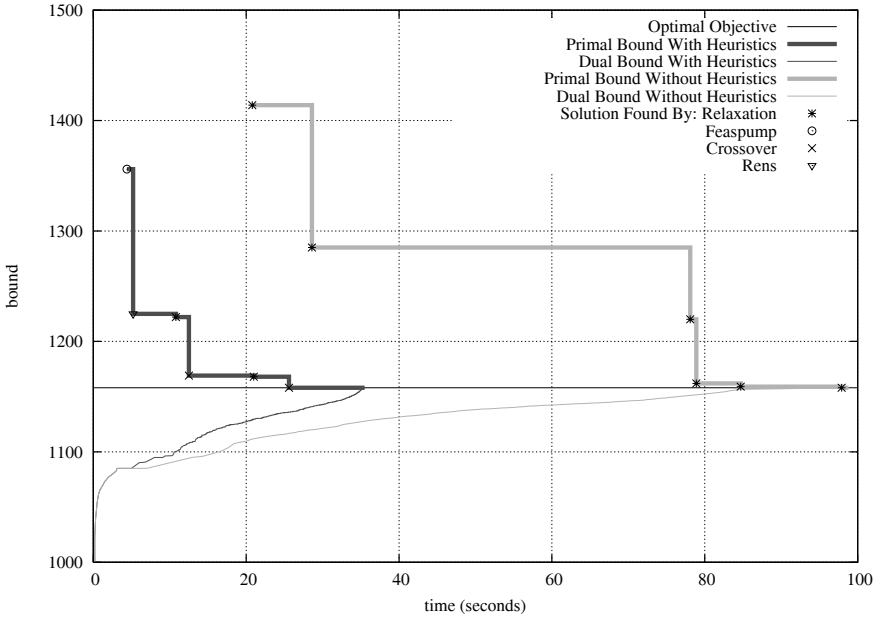


Fig. 1. Instance aflow30a: developing of primal and dual bound if SCIP runs with (dark) and without any heuristics (light)

Figure 1 exemplarily shows the developing of the primal and dual bound for two runs of SCIP 0.82b with an instance taken from the MIPLIB2003 [4]: one with the default heuristics and one without any heuristics activated.

As expected, SCIP with heuristics is faster in finding the first feasible solution, an optimal solution and proving the optimality. We also observe that the dual bound raises faster immediately after feasible solutions were found and that even the first improvement by an integral node LP-relaxation occurs at an earlier step in time. This is due to the fact that with the knowledge of a good primal bound, one is able to prune suboptimal nodes, fix additional variables, which itself leads to stronger cuts and so forth.

All these results emphasize that heuristics are an important part of a branch-cut-and-price-framework and point out the importance of the interaction between different heuristics.

References

1. T. Achterberg. *Constraint Integer Programming*. PhD thesis, Technische Universität Berlin, 2007.
2. T. Achterberg and T. Berthold. Improving the Feasibility Pump. *Discrete Optimization*, Special Issue 4(1):77–86, 2007.
3. T. Achterberg, T. Berthold, M. Pfetsch, and K. Wolter. SCIP (Solving Constraint Integer Programs). `http://scip.zib.de`.
4. T. Achterberg, T. Koch, and A. Martin. MIPLIB 2003. *Operations Research Letters*, 34(4):1–12, 2006. `http://miplib.zib.de`.
5. L. Bertacco, M. Fischetti, and A. Lodi. A feasibility pump heuristic for general mixed-integer problems. *Discrete Optimization*, Special Issue 4(1):77–86, 2007.
6. T. Berthold. Primal Heuristics for Mixed Integer Programs. Master's thesis, Technische Universität Berlin, 2006.
7. T. Berthold. RENS - Relaxation Enforced Neighborhood Search. ZIB-Report 07-28, Konrad-Zuse-Zentrum für Informationstechnik Berlin, 2007. `http://opus.kobv.de/zib/volltexte/2007/1053/`.
8. R. E. Bixby, S. Ceria, C. M. McZeal, and M. W. Savelsbergh. An updated mixed integer programming library: MIPLIB 3.0. *Optima*, (58):12–15, 1998.
9. E. Danna, E. Rothberg, and C. L. Pape. Exploring relaxation induced neighborhoods to improve MIP solutions. *Mathematical Programming A*, 102(1):71–90, 2004.
10. M. Fischetti, F. Glover, and A. Lodi. The feasibility pump. *Mathematical Programming A*, 104(1):91–104, 2005.
11. M. Fischetti and A. Lodi. Local branching. *Mathematical Programming B*, 98(1-3):23–47, 2003.
12. H. Mittelmann. Decision tree for optimization software: Benchmarks for optimization software. `http://plato.asu.edu/bench.html`.

Forecasting Optimization Model of the U.S. Coal, Energy and Emission Markets

Jan-Hendrik Jagla and Lutz Westermann

Institut für Mathematische Optimierung, Carl-Friedrich Gauß Fakultät, Technische Universität Braunschweig, Germany
jan-hendrik@jagla.eu, lutzwestermann@web.de

Generation of electricity is influenced by the complex interplay of economic and political forces, as well as environmental and technological constraints. Sustainable energy-political acting not only has to consider the pure satisfaction of demand and economic interests, but also to accept the challenge of minimizing the environmental impacts. In such a complex system, market participants need reliable knowledge and forecasts in order to act properly. These essential requirements can be obtained through an abstract model providing comprehensive results.

The result of these theses under the supervision of Prof. Dr. Sándor P. Fekete is an advanced model of so far unmatched temporal, regional and physical granularity that minimizes total system costs of the U.S.-American electricity generation, and an efficient and fully operational implementation of the model with the algebraic modeling language GAMS (General Algebraic Modeling System). The implementation entirely satisfies the requirements to be used in practical application by Greenmont Energy Consulting LLC, one of the leading American energy consulting firms.

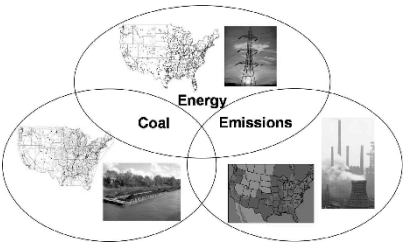


Fig. 1. Integrated view of energy and relevant secondary markets

The optimization model is based on a detailed analysis of the U.S.-American energy market. It turned out that it is not sufficient to only consider the structures of the primary market. Also, the relevant secondary markets, in particular the coal and emissions market, have to be included into the analysis and subsequently into the model.

Since the U.S. possesses large domestic coal reserves, more than half of U.S.-American electricity demand is covered by coal-fired power plants. In other words, the power industry is responsible for more than 90% of the demand for coal. Therefore a comprehensive model must not fail to consider the coal market. Aspects as production and transport as well as the expansion of mine capacities for the different classes and types of coal deserve attention. The term "types of coal" points out that coal is not a homogeneous commodity. The about 100 mined types differ substantially, of course in their costs, but also in energy and pollutant content, grindability, as well as long-term and short-term availability. All of these characteristics are incorporated into the model.

The fact that burning fossil fuels emits several air pollutants makes it clear that the generation of electricity is subject to manifold emission regulations. Regionally and temporally differing emission rules limit the pollutant output, both absolutely and relatively to the amount of electricity generated. Consequently, the approximately 125 electric utilities are encouraged to invest in emission abatement technology for their power plants and to use coal of higher quality. These strategies do not only reduce risks of governmental interventions but also offer economic opportunities in perspective of the increasing relevance of emission allowance trading. Consequently, the trading of emission allowances is regarded as the third substantial market.

The consideration of this complex triad of electricity, coal and emissions market (see Figure 1) establishes the basis for the innovation of this optimization model. In previous models used in consulting practice only parts of the industries and markets involved in electricity generation were represented. Modeling the more complex interaction of the individual markets was possible only by iterative sequential feedback between the separate models, if at all. That way, a great deal of optimization potential remained unused. Where available, existing integrated models renounce the required detailed representation of the actual cause-effect relationships. This work succeeds to model the complex interplay of the various markets holistically and still guarantees high granularity.

The model abandons the option to group identically constructed or geographically close power plants throughout. In order to achieve the

required accuracy, the model treats each of the 2400 plants separately. In case of coal-fired power plants even single boilers are modeled individually in order to pay attention to the high relevance of coal-based electricity generation. Thereby the number of units modeled rises to about 3200. This approach has the advantage that not only technical specifications but also historically originated ecological restrictions can be distinguished. It is precisely this high granularity that allows the model to decide in detail on the installation and use of appropriate technologies to reduce pollutant emissions considering the economic consequences (see Figure 2). Furthermore, the selection of fuels for the individual coal-fired units is not limited artificially as this would anticipate decisions and restrict the solution space. The resulting problem of the determination of coal transportation costs between all mines and all power plants is solved as a separate optimization model based on the rail, water and truck network provided by the U.S. Department of Transportation[1].

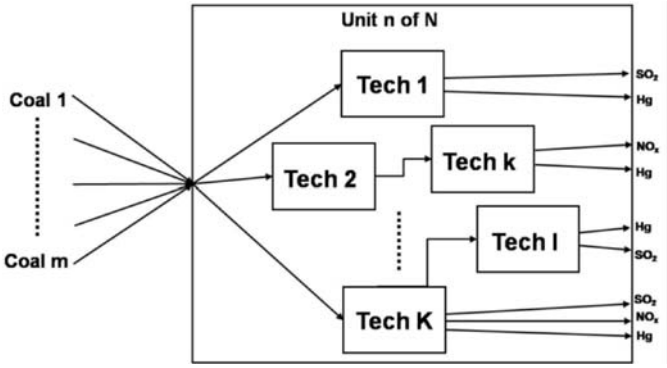


Fig. 2. Modeling emission abatement technologies

Besides the physical granularity, the modeling of different temporal resolutions leads to a better optimization model. It is necessary to be able to make certain decisions on a finer time scale within the general time horizon of one year. Seasonal emission regulations are one example. They imply that decisions about used fuels and level of utilization of clean-up equipment have to be revised several times a year. An even higher temporal granularity is required by the integrated power plant scheduling as it is coupled to the electricity demand depending on the

[1] The data is taken from the National Transportation Atlas Database (NTAD), see http://www.bts.gov/publications/national_transportation_atlas_database/

time of day. Contrariwise, it is possible to extend the model to a multi-year model in order to render long-term forecasts. In doing so, several years are modeled in series and the results of one year are passed into the database of the subsequent one. The key here is the concept of sunk costs which helps to compensate the disadvantage of the lack of foresight.

Due to the resulting size of the optimization problem it is necessary to avoid nonlinearities, for example when modeling emission abatement technologies. An additional difficulty arises from the need for binary variables for the modeling of operational and investment decisions. A first mixed-integer optimization model had a size of 3.5 million constraints and 7.5 million variables (26 million non-zero entries). It turned out that modern optimization tools, such as ILOG CPLEX 10.0, have the ability to solve the problem with a duality gap of less than 0.5% within 13 hours. Nevertheless, neither the runtime nor the high average memory requirements of 6GB are acceptable in practical use since long-term forecasts or scenario analyses require multiple runs. However, we managed to reduce the size of the problem by 80% through a sophisticated hierarchical grouping of coal classes without deviating more than 0.5% from the original solution (see Figure 3). It is highly important to emphasize that this grouping is applied only to the scheduling of emission abatement technologies and that the original granularity aim is not violated. The required computing time is reduced by 95% to less than an hour on average, so that even long-term forecasts of 20 years can be done in acceptable time. The attained solution times as well as the clearly structured and fully adaptable database do not limit the model usage to just the near future.

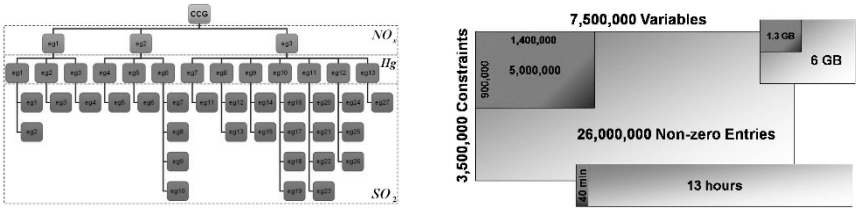


Fig. 3. Hierarchical coal grouping and its impact on the model complexity

In addition to the formulation of the mathematical optimization model of the U.S.-American coal, energy and emission markets the main result of this work is the highly efficient GAMS implementation

GEM™. The primal- and dual solutions of the mixed-integer optimization problem provide comprehensive results as well as highly valuable fine-grained outputs. As can be seen on *www.greenmontenergy.com*, Greenmont Energy Consulting thus has the ability to successfully offer reliable consulting services to such different market participants as mine operators, transportation companies, electric utilities, investors, brokers and government authorities.

Optimal Control Strategies for Incoming Inspection

Stefan Nickel, Sebastian Velten, and Hans-Peter Ziegler

Lehrstuhl für Operations Research und Logistik, Universität des Saarlandes, Campus A5 3, Germany. hp.ziegler@orl.uni-saarland.de

Summary. Acceptance sampling is one important part of logistic processes and production. Because of that, we developed a new model, named DMGI (Dynamic Mixed Goods Inspection) which uses past inspections in order to support companies deciding which delivery should be inspected in goods income.

1 Introduction

Since acceptance sampling is one important part of logistic processes and production, numerous models about optimal inspection plans and sample sizes have been developed to conclude from a small amount of inspected items to the whole delivery. Unfortunately, this concept is not suitable for monitoring the number of goods received.
There are some studies, which allow drawing conclusions regarding the number of articles by weight of the whole delivery or number of palettes. However, a large number of logistic centers get mixtures of different articles, small delivery quantities, irregular weights and packages. So they are not able to determine the number of articles exactly. In this case, companies mostly use some "rules of thumb" for deciding which delivery to inspect and which not. As a result, these strategies are usually far from optimal. To overcome this problem, a new model, named DMGI (Dynamic Mixed Goods Inspection) has been developed which uses past inspections to cluster suppliers into different quality classes. The clustering problem is solved by a p-median on the line using dynamic programming. Moreover a multiobjective, nonlinear program to determine optimal time-periods between two inspections of any supplier subject to his ranking is developed which is solved by dynamic programming, too. DMGI considers stochastic shortages. In addition,

adaptive learning and fast computation times allow a daily adjustment of the decisions. Using simulated data, the new model has been compared with the strategy of a medium-sized enterprise. It is shown, that the number of detected shortfalls are considerably higher using DMGI. This paper presents a new model which uses past observations to determine optimal control strategies for incoming inspections.
The paper is structured as followed:
Section 2 gives an idea for a measure of supplier's quality. Section 3 uses dynamic programming in order to cluster all the suppliers depending on their quality. In Section 4, we present a model that determines optimal time-periods between inspections. Finally, we show some simulation results and give a short summary.

2 A Measure for Supplier's Quality

First of all it is necessary to find a measure for the supplier's quality in order to make them comparable. Since it seems to be expedient to consider the value of the articles as well as the relative shortage of any delivery t of any supplier i, we define the expected loss for not inspecting a delivery for any supplier $\mathrm{E}(V_i)$ as a measure for the supplier's quality.

3 Clustering

Based on this measure, all suppliers should be clustered in k quality-groups. An overview about the most important clustering methods is given in Kaufman and Rousseeuw [7].
We use a k-medoids method. The pursued idea is the following. All the expected disprofits are points on a line which should be clustered in k different groups. Suppose these points represent customers and they should be assigned to k facilities (here our medoids) so that the sum over all distances between the customers and their facilities is minimized. So the arising partitioning around medoids cluster can be solved as a k-median-problem on the line. Because there exists at least one optimal solution, where the facilities are located directly in the points of the customers, this problem can become discretize without changing the optimal solution. Therefor the clustering problem can be written as follows.
Let loc_s $(s = 1, 2, ..., S)$ be the location of the potential facilities.
So the distance between customer $i \in 1, 2, ..., I$ and facility $s \in 1, 2, ..., S$ can be written as $c_{i,s} :=\mid \mathrm{E}(V_i) - loc_s \mid$.

S is at most as large as I because it is possible, because there may exist suppliers with equal expected losses. In such cases there are more than one customer in one point, but it is not reasonable to consider more than one facility there (because our problem is not capacitated).
In addition two types of variables are needed:

$$y_s := \begin{cases} 1, \text{ if facility } s \text{ is opened} \\ 0, \text{ else} \end{cases}$$

and

$$x_{i,s} := \begin{cases} 1, \text{ if customer } i \text{ is assigned to facility } s \\ 0, \text{ else} \end{cases}$$

Using these variables, the model can be written as:

$$\begin{aligned} \text{Min} \sum_{i=1}^{I} \sum_{s=1}^{S} & c_{i,s} \cdot x_{i,s} \\ \text{s.t.} \sum_{s=1}^{S} & y_s \leq k & (1) \\ \sum_{s=1}^{S} & x_{i,s} = 1 \qquad \forall i \in \{1,2,...,I\} & (2) \\ & x_{i,s} \leq y_s \qquad \forall i \in \{1,2,...,I\},\ s \in \{1,2,...,S\} & (3) \\ & y_s \in \{0;1\} \ \forall s \in \{1,2,...,S\} \\ & x_{i,s} \in \mathbb{R}^+ \qquad \forall i \in \{1,2,...,I\},\ s \in \{1,2,...,S\} \end{aligned}$$

The objective minimizes the total distance between the facilities and their customers. Constraints (1) state that no more than k facilities are allowed to be opened. Constraints (2) guarantee every customer to be assigned to exactly one facility. Constraints (3) ensure that any customer can only be assigned to an opened facility.
Note that it is not necessary to define $x_{i,s}$ as an binary variable, since the formulation forces it to zero or one.
Unfortunately this integer problem is $\mathcal{NP}$-hard. Anyhow there exist algorithms, which solve the problem in polynomial time if n is given (see Daskin [2], Kaliv [3], Hassin [5] and Garey [4]). Therefore we solve this problem using dynamic programming. In this way the complexity is $O(I^2 \cdot n)$ and we are able to solve realistic clustering problems with 250 suppliers and 6 groups within 2 seconds.

4 Optimal Time-Period Between Two Inspections

Finally the optimal time-period between two inspections is needed. Therefore we need the assumptions, that all deliveries contain nearly

the same number of articles, in order to make the model as clear as possible. These assumptions can be abandoned easily by extending the program.
There is a limited budget, that only allows inspecting P deliveries in a special time horizon. As from now, we distinguishe between the values of a special supplier and of a special group. For this reason all the variables and parameters become extended by a further index (l for supplier and g for group). So, $V^l_{i,t}$ for example, is the disprofit in the t-th realization of the i-th supplier, while $V^g_{i,t}$ is the t-th realization in group i.
Every uninspected delivery causes a miss and the best estimator for this loss is the expected loss of the previous realizations $\mathrm{E}(V^g_i)$.
Let $A^g_i \in \{1, 2, 3, ...\}$ be the time period between two inspections of any supplier in group i. (Time period denotes in this context the number of deliveries between one inspection and the following, so that $A^g_i = 1$, e.g., means, that every delivery will be inspected, while a time period of two means, that every second will be inspected.)
If the time-period in a group is A^g_i, $\lceil L^g_i / A^g_i \rceil$ of L^g_i deliveries will be inspected.
In order to keep the model as easy as possible, $\lceil L^g_i / A^g_i \rceil$ will be approximated by L^g_i / A^g_i. (This is a really good approximation, if L^g_i is large enough.) So, the expected wastage would be

$$\sum_{i=1}^{n} L^g_i \cdot \frac{A^g_i - 1}{A^g_i} \cdot \mathrm{E}(V^g_i).$$

These observations lead to the following program:

$$\begin{array}{lll}
\text{Min} \sum\limits_{i=1}^{n} L^g_i \cdot \frac{A^g_i - 1}{A^g_i} \cdot \mathrm{E}(V^g_i) & & \\
\text{s.t.} \sum\limits_{i=1}^{n} \frac{L^g_i}{A^g_i} \leq P & & (4) \\
A^g_i \geq 1 & \forall i \in \{1, 2, ..., n\} & (5) \\
A^g_i \in \mathbb{N} & \forall i \in \{1, 2, ..., n\} &
\end{array}$$

The objective minimizes the expected wastage because of uninspected deliveries. Constraints (4) state, that not more than P deliveries are inspected. Constraints (5) ensure that every delivery will be inspected at least once.
In general the number of inspections is much smaller than the number of deliveries, so that unreliable suppliers become inspected almost in every delivery, while reliable suppliers would be inspected very seldom.

Thereby the uncertainty, that a supplier still is in the right group, increases by every uninspected delivery (suppliers could change their quality by changing their employees, their consignment sale or it could be possible that the first observations presented a wrong picture of any supplier...).
Because of that, uncertainty should be incorporated. One well known possibility is using the standard deviation σ_i^g or variance $\mathrm{Var}(V_i^g) = (\sigma_i^g)^2$ (More about measures for uncertainty can bo found in Mulvey [6].). So the variance between two observations is added to the objective, while the rest of the model stays the same:

$$\text{Min} \sum_{i=1}^{n} \alpha \cdot L_i^g \cdot \frac{A_i^g - 1}{A_i^g} \cdot \mathrm{E}(V_i^g) + (1-\alpha) \cdot (A_i^g - 1)^2 \cdot \mathrm{Var}(V_i^g),$$

where α is a weight between 0 and 1 (for more information about multiobjective optimization see Collette and Siarry [1]). As easily can be seen, this program is not linear in A_i^g anymore.
Fortunately this program can be solved with dynamic programming within 2 second for real problem instances.

5 Simulation Results

We simulated shortfalls of 240 suppliers over a time-horizon of 250 days. In doing so, we assumed, that the incidence of receiving a shortfall is binomial distributed. For the size of the shortfall we tested exponential, lognormal, weibull as well as pareto distributions.
We compared our model, DMGI, with a model, used in a German commercial enterprise and obtained the following results:

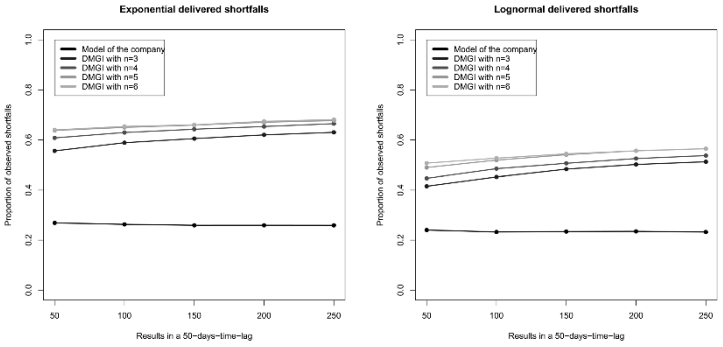


Fig. 1. Results

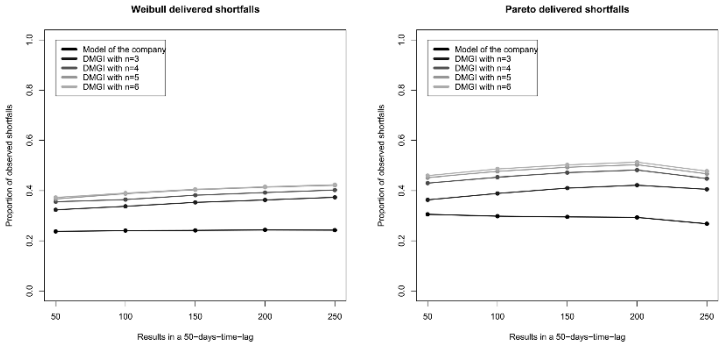


Fig. 2. Results (continued)

6 Conclusions

We have presented a new model, named DMGI, that may support companies deciding which deliveries should be inspected. Our dynamic programming enables the company to adapt its decisions daily, because of the negligible computing times. As the results show, it is possible to observe up to 200% more shortfalls compared to using the selection criteria of a German commercial enterprise.
More details concerning the dynamic programming approach and detailed results can be found in Ziegler [8].

References

1. Y. Collette, P. Siarry. Multiobjective Optimization: Principles and Case Studies. Springer-Verlag. New York. 2003.
2. M. S. Daskin. Network and Discrete Location. Models. Algorithms and Applications. John Wiley & Sons. New York. 1995.
3. O. Kariv, S. L. Hakimi. An Algorithm Approach to Network Location Problems. II: The p-Medians. SIAM Journal on Applied Mathematics 37 (3). 539-560. 1979.
4. M. R. Garey, D. S. Johnson. Computers and Intractability: A Guide to the Theory of NP-Completeness. Freeman. New York. 1979.
5. R. Hassin, A. Tamir. Improved complexity bounds for location problems on the real line. Operations Research Letters. 10:395-402. 1991.
6. J. M. Mulvey, R. J. Vanderbei, S. A. Zentios. Robust Optimization of large-scale systems. Operations Research 43. 264-281. 1995.
7. L. Kaufman, P. J. Rousseeuw. Finding Groups in Data. An Introduction to Cluster Analysis. Wiley Series in Probability and Statistics. John Wiley & Sons. 2005.
8. H.P. Ziegler. Statistische Methoden zur Qualitätskontrolle im Wareneingang bei Handelsunternehmen. 2006.

An Extensive Tabu Search Algorithm for Solving the Lot Streaming Problem in a Job Shop Environment

Liji Shen

Department of Business Administration and Economics, Dresden University of Technology, 01062, Dresden, Germany
liji.shen@mailbox.tu-dresden.de

Summary. The purpose of this paper is to solve the lot streaming problem in job shop scheduling systems, where both equal and consistent sublots are considered. The presented algorithm incorporates a tabu search procedure to determine schedules and a specific heuristic for improving sublot sizes. Computational results confirm that, by applying the lot streaming strategy, production can be significantly accelerated. Moreover, this algorithm yields superior solutions compared to various approaches proposed in the literature and all tested instances show a rapid convergence to their lower bounds.

Key words: Lot Streaming, Job Shop, Tabu Search.

1 Introduction

In this paper I focus on solving the lot streaming problem in a job shop environment, where both equal and consistent sublots are considered. The *job shop* scheduling problem can be briefly described as follows [6]: A set of jobs and a set of machines are given. Each machine can process at most one job at a time. Each job consists of a sequence of operations, which need to be processed during an uninterrupted time period of a given length on a given machine. A *schedule* is an allocation of the operations to time intervals on the machines. The objective is to find a schedule of minimum length (*makespan*). This class of problems is not only NP-hard but also belongs to the most difficult combinatorial optimization problems [8].

With respect to *lot streaming*, a job is actually a *lot* composed of identical items [3]. In classic job shop scheduling systems the entire lot is not transferable before being completed on a machine, which,

however, leads to low machine utilization and long completion time. Lot streaming techniques, in comparison, provide the possibility of splitting a lot into multiple smaller sublots, which can be transported to the next stage upon their completion. As a result of operation overlapping, production can be remarkably accelerated.

Of particular note is that each sublot can be viewed as an individual job and the problem size drastically increases with the total number of sublots. Moreover, the requirement of determining sublot sizes brings additional difficulty in solving the lot streaming problem. Only small instances can be solved employing optimization based software. In order to solve larger instances, I adopt a specific iterative procedure which alternates between the determination of schedules and the (sub)lot-sizing problem [3].

The remainder of the paper is organized as follows: in the subsequent section, the implementation of the fundamental elements as well as some enhancements of tabu search are presented. Section 3 describes the so-called KOL-Heuristic for varying sublot sizes in more detail. Computational results focusing on various aspects are summarized in Section 4.

2 The Tabu Search Implementation

2.1 Neighbourhood Structure

In this neighbourhood only adjacent operations of the same block are observed. In order to intensify the search process, a move may concern three operations [4]. As a result, instead of the simple swap of two operations, a move consists of a sequence of elementary swaps. First, two adjacent operations of the same block are interchanged except when they both are internal. The third operation of the same block, if it exists, is then inserted in relation to the previously arranged two operations.

First of all, this neighbourhood effectively excludes infeasible solutions and a majority of fruitless moves are successfully eliminated. Most importantly, due to the possible insertion of the third operation, this neighbourhood provides a more thorough change to the current schedule and intensifies the search in promising areas.

2.2 Tabu Tenure

In order to properly determine tabu tenure, I incorporate *reactive tabu search* into the algorithm [2]. As an extension, a simple mechanism is

integrated to adjust tabu tenure according to the evolution of the search process. Another important aspect of reactive tabu search consists in its combination with diversification. As illustrated in Figure 1, tabu tenure is actively adapted throughout the search. Moreover, abrupt decrease of both values (tabu tenure and the number of often repeated solutions) indicates the initiation of diversification.

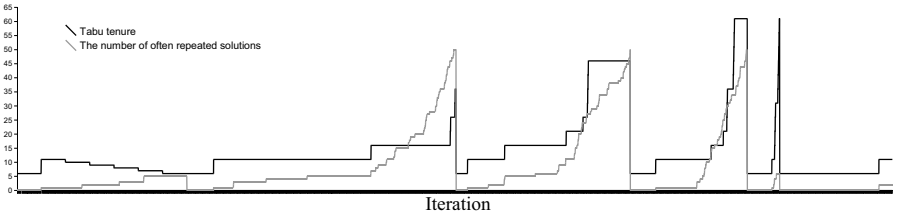


Fig. 1. Reactive Tabu Search

2.3 Diversification

In the algorithm, a simple yet effective method is used to achieve diversity. When diversification is activated, critical operations of the current schedule are first identified. A pair of adjacent critical operations is then arbitrarily selected and interchanged. After sufficient iterations of this procedure, a considerable distance to local optima can be reached.

3 Kol-Heuristic

In this section a specific procedure – the Kol-Heuristic – for solving the sublot-sizing problem is developed. As depicted in Figure 2, it is based on the best known schedule and tests iteratively, if the simultaneous variation of two sublot sizes of the same job can lead to an improved solution.

In contrast to exact methods, the Kol-Heuristic requires negligible computing time. On the other hand, this heuristic also exerts positive influences on the search process. Since the variation of sublot sizes starts from the best known solution, this heuristic can actually be viewed as *intensification*, which represents another key element of tabu search.

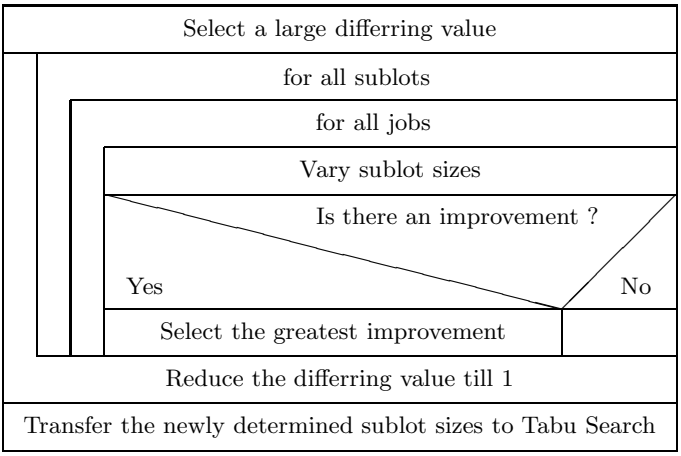


Fig. 2. Framework of the KOL-Heuristic

4 Computational Results

In order to demonstrate the efficiency of the proposed algorithm, computational results are presented in this section. The tested 40 instances are the well-known job shop benchmark problems, which range from 6 jobs on 6 machines to 20 jobs on 15 machines [1, 9, 5].

Since studies on the job shop problem with lot streaming are rather limited, the algorithm is first used to solve standard job shop benchmark problems. More than 80% of the tested instances have reached their global optima. For the remaining instances, the average deviation from optimal solution is less than 1%. Moreover, as shown in Table 1, this algorithm generates superior solutions compared to various tabu search algorithms proposed in the literature [4, 7, 10].

Subsequently, these instances are solved by applying 2 to 4 sublots. The adopted measurement indicates that at least one machine is working without idling time and therefore, this value represents a valid lowest boundary of the problem. It can be seen from Figure 3 that all results converge rapidly to their lower bounds as the number of sublots increases. With 4 sublots, the average deviation from lower bound is only around 1%. This outcome implies that the solutions are already very close to their global optima.

In order to show the performance of the KOL-Heuristic, test results of two famous benchmark instances (ft06, ft10) are presented in Table 2 and compared to those of modified shifting bottleneck procedure [3]. In the case of equal sublots, solutions to ft06 are slightly worse, whereas solutions to ft10 are already better. After sublot sizes being varied

Table 1. Superior solutions compared to different Tabu Search algorithms

Problem size	Instance	Opt.*	Makespan	Nowicki1996¶	Geyik2004‡	Dellamico1993§
10·5	la04	590	590¶‡*	593	598	590
15·5	la06	926	926‡*	926	936	926
	la07	890	890‡*	890	910	890
	la10	958	958‡*	958	1034	958
20·5	la13	1150	1150‡*	1150	1159	1150
	la14	1292	1292‡*	1292	1374	1292
10·10	la16	945	945¶‡*	946	959	945
	la18	848	848‡*	848	861	848
	la19	842	842‡*	842	860	842
	la20	902	902‡*	902	909	902
	abz05	1234	1236¶‡	1238	1238	1236
	abz06	943	943¶‡*	945	947	943
	ft10	930	930‡*	930	971	930
15·10	la22	927	930¶‡§	954	962	933
	la24	935	941¶‡	948	989	941
	la25	977	978¶‡§	988	995	979
20·10	la26	1218	1218‡*	1218	1240	1218
	la28	1216	1216‡*	1216	1221	1216
15·15	la36	1268	1268¶‡§*	1275	1302	1278
	la37	1397	1409¶‡	1422	1453	1409
	la39	1233	1238‡§	1235	1269	1242
	la40	1222	1228¶‡§	1234	1261	1233

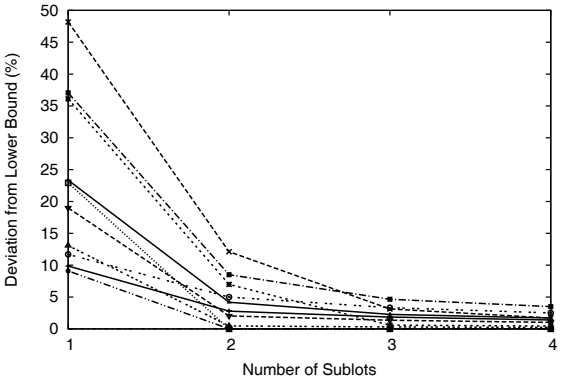


Fig. 3. Solutions to the instances with lot streaming

with the KOL-Heuristic, all results are further improved and superior to those of shifting bottleneck procedure.

Table 2. Solutions to Ft06 and Ft10 with sublots

		Ft06 The number of sublots				Ft10 The number of sublots			
Algorithm	Sublot Type	1	2	3	4	1	2	3	4
Tabu Search	equal	55,00	47,50	45,33	44,50	930,00	736,50	674,00	651,75
	consistent	55,00	45,67	44,00	43,40	930,00	732,62	673,67	649,57
Shifting Bottleneck	consistent	55,00	46,19	44,31	43,41	950,00	776,64	696,60	672,88

References

1. Adams J, Balas E, Zawack D (1998) The shifting bottleneck procedure for job shop scheduling. Management Science 34: 391–401
2. Battiti R, Giampietro T (1994) The reactive tabu search. ORSA Journal on Computing 6(2): 126–140
3. Dauzère-Pérès S, Lasserre J (1997) Lot streaming in job-shop scheduling. Operations Research 45:584–595
4. Dell'amico M, Trubian M (1993) Applying tabu search to the job-shop problem. Annals of Operations Research 41: 231–252
5. Fisher H, Thompson GL (1963) Probabilistic learning combinations of local job-shop scheduling rules. In: Muth JF, Thompson GL (eds) Industrial Scheduling. Prentice Hall, Englewood Cliffs, New Jersey
6. French S (1982) Sequencing and scheduling: an introduction to the mathematics of the job-shop. Horwood, Chichester, U.K.
7. Geyik F, Cedimoglu IH (2004) The strategies and parameters of tabu search for job-shop scheduling. Journal of Intelligent Manufacturing 15: 439–448
8. Lawler EL, Lenstra JK, Rinnooy Kan AHJ (1982) Recent developments in deterministic sequencing and scheduling: A survey. In: Dempster MSH, Lenstra JK, Rinnooy Kan AHJ (eds) Deterministic and stochastic scheduling. Reichel, Dordrecht, The Netherlands
9. Lawrence S (1984) Resource constrained project scheduling: An experimental investigation of heuristic scheduling techniques (supplement). Graduate School of Industrial Administration, Carnegie-Mellon University, Pittsburgh, PA
10. Nowicki E, Smutnicki C (1996) A fast taboo search algorithm for the job shop problem. Management Science 42(6): 797–813

Part III

Applied Probability and Stochastic Programming

Optimizing Consumption and Investment: The Case of Partial Information

Markus Hahn, Wolfgang Putschögl, and Jörn Sass

RICAM, Austrian Academy of Sciences, Altenberger Str. 69, A-4040 Linz, Austria. markus.hahn@oeaw.ac.at, wolfgang.putschoegl@oeaw.ac.at, joern.sass@oeaw.ac.at

1 Introduction

In Section 2 we present a stock market model where prices satisfy a stochastic differential equation with a stochastic drift process which is independent of the driving Brownian motion. The investor's objective is to maximize the expected utility of consumption and terminal wealth under partial information, meaning that investment decisions are based on the knowledge of the stock prices only, cf. [2, 3]. Consumption and investment processes as well as the optimization problem are introduced in Section 3. Optimal consumption and optimal terminal wealth can be expressed in terms of the filter for the Radon Nikodym density of the risk neutral probability under which the price processes become martingales. The solution to this classical optimization problem is provided in Section 4 where consumption and investment strategies are computed based on Malliavin derivatives of the corresponding density process. The results apply to both classical models for the drift process, a linear Gaussian model (GD) and a continuous time Markov chain (HMM). In Section 5 and 6 we look at these two cases, and show that they satisfy all the conditions for the optimal strategies, see also [3, 5]. For proofs and further details we refer to [4] if not mentioned differently. In addition to [4] we compare in Section 7 the HMM with the GD model when applied to historical prices. For parameter estimation we use a modification of the MCMC methods derived in [1].

Notation. The symbol $^\top$ will denote transposition, $\mathrm{Diag}(v)$ is the diagonal matrix with diagonal v, Id_n denotes the n-dimensional identity matrix, and $\mathcal{F}^X = (\mathcal{F}_t^X)_{t\in[0,T]}$ stands for the filtration of augmented σ-algebras generated by the $\mathcal{F}$-adapted process $X = (X_t)_{t\in[0,T]}$.

2 The Basic Model

Let $(\Omega, \mathcal{G}, \mathrm{P})$ be a complete probability space, $T > 0$ the terminal trading time, and $\mathcal{F} = (\mathcal{F}_t)_{t\in[0,T]}$ a filtration in $\mathcal{G}$ satisfying the usual conditions. We can invest in a money market (bond) and n stocks. The bond $(S^0_t)_{t\in[0,T]}$ has constant interest rate r. For $S^0_0 = 1$ we have $S^0_t = \exp(rt)$ and the corresponding *discount factor* reads $\beta_t = \exp(-rt)$. The *price process* $S = (S_t)_{t\in[0,T]}$, $S_t = (S^1_t, \dots, S^n_t)^\top$, of the stocks evolves like

$$\mathrm{d}S_t = \mathrm{Diag}(S_t)(\mu_t\,\mathrm{d}t + \sigma\,\mathrm{d}W_t)\;, \quad S_0 = s_0\;,$$

where $W = (W_t)_{t\in[0,T]}$ is a n-dimensional standard Brownian motion with respect to P. We assume that the drift vector $\mu \in \mathbb{R}^n$ is adapted to the filtration $\mathcal{F}$ and the volatility matrix $\sigma \in \mathbb{R}^{n\times n}$ is constant and non-singular. The *return process* $R = (R_t)_{t\in[0,T]}$ associated with the stocks is defined by $\mathrm{d}R_t = \mathrm{Diag}(S_t)^{-1}\,\mathrm{d}S_t$ and the *excess return process* $\tilde{R} = (\tilde{R}_t)_{t\in[0,T]}$ by $\mathrm{d}\tilde{R}_t = \mathrm{d}R_t - r\mathbf{1}_n\,\mathrm{d}t = (\mu_t - r\mathbf{1}_n)\,\mathrm{d}t + \sigma\,\mathrm{d}W_t$. We define the market price of risk $\theta = (\theta_t)_{t\in[0,T]}$ by $\theta_t = \sigma^{-1}(\mu_t - r\mathbf{1}_n)$, and the density process $Z = (Z_t)_{t\in[0,T]}$ by $\mathrm{d}Z_t = -Z_t\theta_t^\top\,\mathrm{d}W_t$.

Assumption 1 *Suppose that* $\int_0^T \|\theta_t\|^2\,\mathrm{d}t < \infty$ *a.s. and that* Z *is a martingale under* P *w.r.t. the filtration* $\mathcal{F}$.

We consider the case of *partial information*, i.e., we can only observe the stock prices. Thus only events of $\mathcal{F}^S$ are observable and the strategy has to be adapted to $\mathcal{F}^S$.

The *conditional density* $\zeta_t = \mathrm{E}[Z_t|\mathcal{F}^S_t]$ and its discounted version $\tilde{\zeta}_t = \beta_t\zeta_t$ will be crucial for filtering and optimization. Let $\hat{\mu}_t = \mathrm{E}[\mu_t|\mathcal{F}^S_t]$ denote the filter for μ_t. Next, we introduce the *risk neutral probability measure* $\tilde{\mathrm{P}}$ by $\mathrm{d}\tilde{\mathrm{P}} = Z_T\,\mathrm{dP}$. We denote by $\tilde{\mathrm{E}}$ expectation under $\tilde{\mathrm{P}}$. Girsanov's theorem guarantees that $\mathrm{d}\tilde{W}_t = \mathrm{d}W_t + \theta_t\,\mathrm{d}t$ defines a $\tilde{\mathrm{P}}$-Brownian motion w.r.t. $\mathcal{F}$. Thus, also the excess return process $\mathrm{d}\tilde{R}_t = (\mu_t - r)\,\mathrm{d}t + \sigma\,\mathrm{d}W_t = \sigma\,\mathrm{d}\tilde{W}_t$ is a martingale under $\tilde{\mathrm{P}}$.

3 Consumption and Investment Processes

A *consumption process* $c = (c_t)_{t\in[0,T]}$ is a nonnegative, one-dimensional, $\mathcal{F}^S$-adapted, measurable process satisfying $\int_0^T c_u\,\mathrm{d}u < \infty$ a.s. An *investment process* $\pi = (\pi_t)_{t\in[0,T]}$ is a n-dimensional $\mathcal{F}^S$-adapted, measurable process satisfying $\int_0^T \left(|\pi_u^\top\mu_u| + \|\pi_u^\top\sigma\|^2\right)\mathrm{d}u < \infty$ a.s. Here, c_t

and π_t represent the *rate of consumption* and the *wealth invested in the stocks*, respectively, at time t. For *initial capital* $x_0 > 0$ the *wealth process* $X^{c,\pi} = (X_t^{c,\pi})_{t\in[0,T]}$ corresponding to the consumption/investment pair (c, π) is well defined and satisfies

$$\mathrm{d}X_t^{c,\pi} = \pi_t^\top(\mu_t\,\mathrm{d}t + \sigma\,\mathrm{d}W_t) + (X_t^{c,\pi} - \mathbf{1}_n^\top\pi_t)r\,\mathrm{d}t - c_t\,\mathrm{d}t\ , \quad X_0^{c,\pi} = x_0\ .$$

A consumption/investment pair (c, π) is called *admissible* for initial capital $x_0 > 0$ if $X_t^{c,\pi} \geq 0$ a.s. for all $t \in [0, T]$. We denote the class of admissible (c, π) for initial capital x_0 by $\mathcal{A}(x_0)$.

A *utility function* $U\colon [0,\infty) \to \mathbb{R}\cup\{-\infty\}$ is strictly increasing, strictly concave, twice continuously differentiable, and satisfies $\lim_{x\to\infty} U'(x) = 0$ and $\lim_{x\downarrow 0} U'(x) = \infty$. Further, I denotes the inverse function of U'.

Assumption 2 *We demand that I satisfies $I(y) \leq Ky^a$, $|I'(y)| \leq Ky^{-b}$ for all $y \in (0,\infty)$ and some positive constants a, b, K.*

Well known examples for utility functions are the logarithmic utility function $U(x) = \log(x)$ and the power utility function $U(x) = x^\alpha/\alpha$ for $\alpha < 1, \alpha \neq 0$.

Optimization Problem. For given initial capital $x_0 > 0$ and utility functions U_1, U_2 we consider the classical problem of maximizing the utility from both consumption and terminal wealth, i.e.,

$$\text{maximize } \mathrm{E}\left[\int_0^T \gamma_t U_1(c_t)\,\mathrm{d}t + \gamma_T U_2(X_T)\right] \text{ over } (c,\pi) \in \mathcal{A}(x_0)$$

under the constraint $\mathrm{E}\left[\int_0^T \gamma_t U_1^-(c_t)\,\mathrm{d}t + U_2^-(X_T)\right] < \infty$, where the discount factor $\gamma = (\gamma_t)_{t\in[0,T]}$ is deterministic.

4 Optimization

We introduce the function $\mathcal{X}\colon (0,\infty) \mapsto (0,\infty]$ by

$$\mathcal{X}(y) = \mathrm{E}\left[\int_0^T \tilde{\zeta}_t I_1(y\gamma_t^{-1}\tilde{\zeta}_t)\,\mathrm{d}t + \tilde{\zeta}_T I_2(y\gamma_T^{-1}\tilde{\zeta}_T)\right].$$

In the following D_t will denote the Malliavin derivative. For a definition of the spaces $\mathbb{D}_{p,1}$ and an overview of the results concerning Malliavin calculus we need for our purpose we refer to [4].

Theorem 1. *Suppose that $\mathcal{X}(y) < \infty$ for every $y \in (0,\infty)$. Then there exists a unique number $y^* \in (0,\infty)$ such that $\mathcal{X}(y^*) = x_0$. If*

(1) for some $p, q > 1, \frac{1}{p} + \frac{1}{q} = 1$, *and for all* $s \in [0,T]$ $\tilde{\zeta}_s \in \mathbb{D}_{p,1}$ *and* $I_1'(y^*\gamma_s^{-1}\tilde{\zeta}_s) \in L^q(\tilde{\mathrm{P}})$,
(2) for some $r > 1$ $\sup_{s\in[0,T]} \tilde{\mathrm{E}}[|\beta_s I_1(y^*\gamma_s^{-1}\tilde{\zeta}_s)|^r] < \infty$,
(3) $\sup_{s\in[0,T]} \tilde{\mathrm{E}}\Big[\int_0^T \|\beta_s I_1'(y^*\gamma_s^{-1}\tilde{\zeta}_s) y^*\gamma_s^{-1} D_t\tilde{\zeta}_s\|^4 \,\mathrm{d}t\Big] < \infty$,
(4) $s \mapsto D_t(\beta_s I_1(y^*\gamma_s^{-1}\tilde{\zeta}_s))$ *is continuous on* $(t,T]$ *for almost every* $(t,\omega) \in [0,T]\times\Omega$,

then the optimal consumption process and terminal wealth are given by

$$c_t^* = I_1(y^*\gamma_t^{-1}\tilde{\zeta}_t)\,, \quad X_T^* = I_2(y^*\gamma_T^{-1}\tilde{\zeta}_T)\,,$$

and the unique optimal trading strategy is given by

$$\pi_t^* = \beta_t^{-1}(\sigma^\top)^{-1}\tilde{\mathrm{E}}\bigg[\int_t^T \beta_u I_1'(y^*\gamma_t^{-1}\tilde{\zeta}_u) y^*\gamma_u^{-1} D_t\tilde{\zeta}_u \,\mathrm{d}u + \beta_T I_2'(y^*\gamma_T^{-1}\tilde{\zeta}_T) y^*\gamma_T^{-1} D_t\tilde{\zeta}_T \;\Big|\; \mathcal{F}_t^S\bigg]\,.$$

5 Gaussian Dynamics (GD) for the Drift

In this section we model the drift as in [3] as the solution of the stochastic differential equation $\mathrm{d}\mu_t = \kappa(\bar{\mu} - \mu_t)\,\mathrm{d}t + \upsilon\,\mathrm{d}\bar{W}_t$, where $\bar{W}$ is a n-dimensional Brownian motion w.r.t. $(\mathcal{F}, \mathrm{P})$, independent of W under P, and $\kappa, \upsilon \in \mathbb{R}^{n\times n}$, $\bar{\mu} \in \mathbb{R}^n$. We assume that υ is non-singular and that μ_0 follows a n-dimensional normal distribution with known mean vector $\hat{\mu}_0$ and covariance matrix ϱ_0. Under some conditions on the drift parameters Assumption 1 is satisfied; further we have to strengthen Assumption 2 to a version which is still valid for a wide class of utility functions, e.g. for power utility $U(x) = x^\alpha/\alpha$ with $\alpha < 0.2$, cf. [3, 4].

We are in the situation of *Kalman-Bucy filtering* with signal μ and observation R, and the conditional mean $\hat{\mu}_t = \mathrm{E}[\mu_t \mid \mathcal{F}_t^S]$ satisfies

$$\hat{\mu}_t = \chi_t\Big[\hat{\mu}_0 + \int_0^t \chi_s^{-1}\varrho_s(\sigma\sigma^\top)^{-1}\,\mathrm{d}R_s + \int_0^t \chi_s^{-1}\,\mathrm{d}s\;\kappa\bar{\mu}\Big]\,,$$

where $\dot{\chi}_t = \Big[-\kappa - \varrho_t(\sigma\sigma^\top)^{-1}\Big]\chi_t, \chi_0 = \mathrm{Id}_n$ and $\dot{\varrho}_t = -\varrho_t(\sigma\sigma^\top)^{-1}\varrho_t - \kappa\varrho_t - \varrho_t\kappa^\top + \upsilon\upsilon^\top$. For an explicit solution in the case $n = 1$ we refer to [3, p. 84]. The process $\zeta^{-1} = (\zeta_t^{-1})_{t\in[0,T]}$ satisfies the SDE $\mathrm{d}\zeta_t^{-1} = \zeta_t^{-1}(\hat{\mu}_t - r\mathbf{1}_n)^\top(\sigma^\top)^{-1}\,\mathrm{d}\tilde{W}_t$. We shall write ζ^{GD} for ζ as introduced in this section.

Lemma 1. *For* ζ^{GD} *conditions (1)–(4) of Theorem 1 are satisfied.*

6 A Hidden Markov Model (HMM) for the Drift

In this section we model the *drift process* μ of the return as a *continuous time Markov chain* given by $\mu_t = BY_t$, where $B \in \mathbb{R}^{n\times d}$ is the *state matrix* and Y is a *continuous time Markov chain* with *state space* $\{e_1, \dots, e_d\}$, the standard unit vectors in $\mathbb{R}^d$. The *state process* Y is further characterized by its *rate matrix* $Q \in \mathbb{R}^{d\times d}$, where $Q_{kl} = \lim_{t\to 0} \frac{1}{t} \mathrm{P}(Y_t = e_l | Y_0 = e_k), k \neq l$, is the jump rate or transition rate from e_k to e_l. Moreover, $\lambda_k = -Q_{kk} = \sum_{l=1,l\neq k}^{d} Q_{kl}$ is the rate of leaving e_k. Therefore, the waiting time for the next jump is exponentially distributed with parameter λ_k, and Q_{kl}/λ_k is the probability that the chain jumps to e_l when leaving e_k for $l \neq k$.

The market price of risk becomes $\theta_t = \Theta Y_t$, $\Theta = \sigma^{-1}(B - r\mathbf{1}_{n\times d})$. Hence, the density process Z satisfies $\mathrm{d}Z_t = -Z_t(\Theta Y_t)^\top \mathrm{d}W_t$. Since $(\Theta Y_t)_{t\in[0,T]}$ is bounded, Novikov's condition ensures that Z is a martingale and Assumption 1 is satisfied. Note that $\tilde{\mathrm{P}}$ defined as in Section 2 is also used for the filtering and that $\mathcal{F}^S = \mathcal{F}^R = \mathcal{F}^{\tilde{W}}$. Hence, we are in the situation of *HMM filtering* with signal Y and observation R. The *normalized filter* $\hat{Y}_t = \mathrm{E}[Y_t|\mathcal{F}_t^S]$ and the conditional density $\zeta_t = \mathrm{E}[Z_t|\mathcal{F}_t^S]$ can be expressed in terms of the *unnormalized filter* $\mathcal{E}_t = \tilde{\mathrm{E}}[Z_T^{-1} Y_t|\mathcal{F}_t^S]$ which satisfies

$$\mathcal{E}_t = \mathrm{E}[Y_0] + \int_0^t Q^\top \mathcal{E}_s \,\mathrm{d}s + \int_0^t \mathrm{Diag}(\mathcal{E}_s)\Theta^\top \,\mathrm{d}\tilde{W}_s \,, \quad t \in [0,T] \,.$$

The normalized filter is given by $\hat{Y}_t = \zeta_t \mathcal{E}_t$ and $\zeta_t^{-1} = \mathbf{1}_d^\top \mathcal{E}_t$.

Lemma 2. *For $\zeta^{HMM} = \zeta$ conditions (1)–(4) of Theorem 1 are satisfied.*

7 Numerical Example

We consider daily prices for 20 stocks of the Dow Jones Industrial Index for 30 years, 1972–2001, and the corresponding historic fed rates. For each stock we use parameter estimates for the HMM based on a Markov chain Monte Carlo method, cf. [1]. Also the parameters for GD are obtained from a multiple-block MCMC sampler based on time discretization similar to the sampler described in [1]. The parameter estimates are based on five years and in the subsequent year the strategy is computed. We start with initial capital $X_0 = 1$. We compare the strategy based on the HMM and the GD with the Merton-strategy resulting from the assumption of a constant drift, where we include

Table 1. Application to historical data

	$U_1(x) = U_2(x) = \log(x)$					$U_1(x) = U_2(x) = -x^{-5}/5$		
	HMM	HMM[0, 1]	GD	GD[0, 1]	Const. μ	HMM	GD	Const. μ
$\int_0^T \gamma_t U(c_t^*)\,\mathrm{d}t + \gamma_T U(X_T^*)$								
mean	-1.097	-1.117	-1.620	-1.202	-1.211	-10.407	-49.065	-9.282
$\int_0^T \gamma_t c_t^*\,\mathrm{d}t + \gamma_T X_T^*$								
mean	1.805	1.067	0.478	1.017	1.070	1.046	0.947	1.018
bankrupt	14	0	73	0	2	0	1	0

bankruptcies with utility -1.

The numerical results in Table 1 highlight the two problems we face when applying results of continuous time optimization to market data: model and discretization errors. The 14 bankruptcies for the HMM mainly fall on the Black Monday 1987, where single stocks had losses up to 30%. Even without these jumps extreme long and short positions can lead to high losses when trading only daily. For GD the positions are even more extreme, since the drift process is unbounded, leading to a very poor performance. A rigorous possibility to reduce the risk is imposing constraints on the strategy. Using rectangular constraints (no borrowing/short-selling: $\eta_t \in [0, 1]$) the HMM[0, 1] and GD[0, 1] clearly outperform the Merton-strategy in expected utility. This seems to be more reasonable than using very risk averse power utilities as e.g. with $\alpha = -5$ in Table 1. These also reduce the extreme positions but punish a poor performance such heavily that the average utility is dominated by a few outliers, even if no bankruptcy occurs.

References

1. Hahn M, Frühwirth-Schnatter S, Sass J (2007) Markov chain Monte Carlo methods for parameter estimation in multidimensional continuous time Markov switching models, RICAM-Report No. 2007-09
2. Lakner P (1995) Utility maximization with partial information, Stochastic Process. Appl. 56:247–273
3. _____ (1998) Optimal trading strategy for an investor: the case of partial information, Stochastic Process. Appl. 76:77–97
4. Putschögl W, Sass J (2007) Optimal consumption and investment under partial information, Preprint
5. Sass J, Haussmann UG (2004) Optimizing the terminal wealth under partial information: The drift process as a continuous time Markov chain, Finance Stoch. 8:553–577.

Multistage Stochastic Programs via Stochastic Parametric Optimization

Vlasta Kaňková

Institute of Information Theory and Automation
Academy of Sciences of the Czech Republic
Pod Vodárenskou věží 4, 182 08 Praha 8, Czech Republic
kankova@utia.cas.cz

1 Introduction

Multistage stochastic programming problems can be defined as a finite system of (mostly parametric) one–stage stochastic programming problems with an inner type of dependence (for details see e.g. [1], [2], [6]). Employing this approach we can introduce the multistage ($M+1$–stage, $M \geq 1$) stochastic programming problem as the problem.

Find

$$\varphi_{\mathcal{F}}(M) = \inf \{\mathsf{E}_{F^{\xi^0}} g^0_{\mathcal{F}}(x^0, \xi^0) |\ x^0 \in \mathcal{K}^0\}, \tag{1}$$

where the function $g^0_{\mathcal{F}}(x^0, z^0)$ is defined for $k = 0, 1, \ldots, M-1$ recursively

$$\begin{aligned} g^k_{\mathcal{F}}(\bar{x}^k, \bar{z}^k) \quad &= \\ &\inf\{\mathsf{E}_{F^{\xi^{k+1}|\bar{\xi}^k = \bar{z}^k}}\ g^{k+1}_{\mathcal{F}}(\bar{x}^{k+1}, \bar{\xi}^{k+1})\, | x^{k+1} \in \mathcal{K}^{k+1}_{\mathcal{F}}(\bar{x}^k, \bar{z}^k)\}, \\ g^M_{\mathcal{F}}(\bar{x}^M, \bar{z}^M) &:= g^M_0(\bar{x}^M, \bar{z}^M), \quad \mathcal{K}_0 := X^0. \end{aligned} \tag{2}$$

$\xi^j := \xi^j(\omega),\ j = 0, 1, \ldots, M$ denotes an s–dimensional random vector defined on a probability space $(\Omega, \mathcal{S}, P)$; $F^{\xi^j}(z^j),\ z^j \in R^s,\ j = 0, 1 \ldots, M$ the distribution function of the ξ^j and $F^{\xi^k|\bar{\xi}^{k-1}}(z^k|\bar{z}^{k-1})$, $z^k \in R^s,\ \bar{z}^{k-1} \in R^{(k-1)s},\ k = 1, \ldots, M$ the conditional distribution function (ξ^k conditioned by $\bar{\xi}^{k-1}$); $P_{F^{\xi^j}},\ P_{F^{\xi^{k+1}|\bar{\xi}^k}},\ j = 0, 1, \ldots, M,\ k = 0, 1, \ldots, M-1$ the corresponding probability measures; $Z^j := Z_{F^{\xi^j}} \subset R^s,\ j = 0, 1, \ldots, M$ the support of the probability measure $P_{F^{\xi^j}}$. Furthermore, the symbol $g^M_0(\bar{x}^M, \bar{z}^M)$ denotes a continuous function defined on $R^{n(M+1)} \times R^{s(M+1)};\ X^k \subset R^n,\ k = 0, 1, \ldots, M$ is a nonempty

set; the symbol $\mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k,\, \bar{z}^k) := \mathcal{K}_{F^{\xi^{k+1}|\bar{\xi}^k}}^{k+1}(\bar{x}^k,\, \bar{z}^k),\ k = 0,\, 1,\, \ldots,\, M-1$ denotes a multifunction mapping $R^{n(k+1)} \times R^{s(k+1)}$ into the space of subsets of R^n. $\bar{\xi}^k(:= \bar{\xi}^k(\omega)) = [\xi^0,\, \ldots,\, \xi^k];\ \bar{z}^k = [z^0,\, \ldots,\, z^k],\ z^j \in R^s;\ \bar{x}^k = [x^0,\, \ldots,\, x^k],\ x^j \in R^n;\ \bar{X}^k = X^0 \times X^1 \ldots \times X^k;\ \bar{Z}^k := \bar{Z}_{\mathcal{F}}^k = Z_{F^{\xi^0}} \times Z_{F^{\xi^1}} \ldots \times Z_{F^{\xi^k}},\ j = 0,\, 1,\, \ldots, k,\ k = 0,\, 1,\, \ldots,\, M$. Symbols $\mathsf{E}_{F^{\xi^0}},\ \mathsf{E}_{F^{\xi^{k+1}|\bar{\xi}^k=\bar{z}^k}},\ k = 0,\, 1,\, \ldots,\, M-1$ denote the operators of mathematical expectation corresponding to $F^{\xi^0},\ F^{\xi^{k+1}|\bar{\xi}^k=\bar{z}^k}$.

The definition of the multistage stochastic programs (1), (2) can be "suitable" (see e.g. [6], [7], [11]) for a stability investigation, scenario construction as well as for the investigation of empirical estimates. However, first, it is necessary to state assumptions guaranteing "necessary" properties of the individual problems. To this end, we assume.

A.1 $\{\xi^k\}_{k=-\infty}^{\infty}$ follows a nonlinear autoregressive sequence

$$\xi^k = H(\xi^{k-1}) + \varepsilon^k, \tag{3}$$

where $\xi^0,\ \varepsilon^k,\ k = 1,\, 2,\, \ldots$ are stochastically independent; $\varepsilon^k,\ k = 1,\, \ldots,$ identically distributed. $H := (H_1,\, \ldots,\, H_s)$ is a Lipschitz vector function on R^s (we denote the distribution function of $\varepsilon^1 = (\varepsilon_1^1,\, \ldots,\, \varepsilon_s^1)$ by F^{ε} and suppose the realization of ξ^0 to be known),

A.2 there exist functions $f_{i,\,j}^{k+1},\ i = 1,\, \ldots,\, s,\ j = 1,\, \ldots,\, k+1,\ k = 0,\, \ldots,\, M-1$ defined on R^n and $\alpha_i \in (0,\, 1),\ i = 1,\, \ldots,\, s,\ \bar{\alpha} = (\alpha_1,\, \ldots,\, \alpha_s)$ such that

$$\mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k,\, \bar{z}^k) \quad (:= \mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k,\, \bar{z}^k;\, \bar{\alpha})) \quad =$$

$$\bigcap_{i=1}^{s} \{x^{k+1} \in X^{k+1} :\, P_{F^{\xi^{k+1}|\bar{\xi}^k=\bar{z}^k}}\{\sum_{j=1}^{k+1} f_{i,\,j}^{k+1}(x^j) \le \xi_i^{k+1}\} \ge \alpha_i\}, \tag{4}$$

$$\xi^{k+1} = (\xi_1^{k+1},\, \ldots,\, \xi_s^{k+1}).$$

A very similar case has been investigated in [9]. However, there were not studied assumptions to be constraints sets of the problems (2) nonempty. Of course, there are known (from the stochastic programming literature) sufficient assumptions guaranteing this property in the linear case (fixed complete recourse matrices) or generally relatively complete recourse constraints (for more details see [1]). We try to extend a class of assumptions guaranteing this property. To this end we employ the approach introduced in [10]. Furthermore, we mention assumptions guaranteing to be individual objective functions finite. At the end, we summarize the introduced assumptions to stability results suitable for constructions of approximate solution schemes.

2 Problem Analysis

The multistage problem (1), (2) is (from the probability point of view) under the assumption A.1 determined by the distribution functions F^{ξ^0}, F^{ε}. Employing this fact we can obtain a new relation for $\mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k, \bar{z}^k)$, $k = 0, 1, ..., M-1$. To this end we define $k_{F_i^{\xi^{k+1}|\bar{\xi}^k=\bar{z}^k}}(\alpha_i)$, $k_{F_i^{\varepsilon}}(\alpha_i)$, $\alpha_i \in (0, 1)$, $i = 1, \ldots, s$, $\quad k = 0, 1, \ldots, M-1$ by

$$\begin{aligned} k_{F_i^{\xi^{k+1}|\xi^k=z^k}}(\alpha_i) &= \sup_{z_i^{k+1}\in R^1} P_{F_i^{\xi^{k+1}|\bar{\xi}^k=\bar{z}^k}}\{z_i^{k+1} \le \xi_i^{k+1}\} \ge \alpha_i\}, \\ k_{F_i^{\varepsilon}}(\alpha_i) &= \sup_{z_i\in R^1} P_{F_i^{\varepsilon}}\{z_i \le \varepsilon_i\} \ge \alpha_i\}. \end{aligned}$$

$F_i^{\xi^{k+1}|\bar{\xi}^k=\bar{z}^k}$, F_i^{ε}, $i = 1, \ldots, s$ are corresponding one–dimensional marginal distribution functions. Since (under A.1) $k_{F_i^{\varepsilon}}(\alpha_i) = k_{F_i^{\xi^{k+1}|\bar{\xi}^k=\bar{z}^k}}(\alpha_i) - H_i(z^k)$, we can under A.1, A.2 obtain

$$\begin{aligned} \mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k, \bar{z}^k) &= \bigcap_{i=1}^{s}\{x^{k+1}\in X^{k+1} : \sum_{j=1}^{k+1} f_{i,j}^{k+1}(x^{k+1}) \le k_{F_i^{\xi^{k+1}|\xi^k=z^k}}(\alpha_i)\}, \\ &= \bigcap_{i=1}^{s}\{x^{k+1}\in X^{k+1} : \sum_{j=1}^{k+1} f_{i,j}^{k+1}(x^{k+1}) \le k_{F_i^{\varepsilon}}(\alpha_i) + H_i(z^k)\}\}. \end{aligned} \tag{5}$$

Defining (for given $\bar{\alpha}$) $h_i^{k+1}(\bar{x}^k, \bar{z}^k)$, $i = 1, \ldots, s$, $k = 0, \ldots, M-1$ by

$$h_i^{k+1}(\bar{x}^k, \bar{z}^k) := h_i^{k+1}(\bar{x}^k, \bar{z}^k, k_{F_i^{\varepsilon}}(\alpha_i)) = k_{F_i^{\varepsilon}}(\alpha_i) + H_i(z^k) - \sum_{j=1}^{k} f_{i,j}^{k+1}(x^j), \tag{6}$$

we obtain "classical nonlinear" constraints sets in the form

$$\mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k, \bar{\xi}^k) = \bigcap_{i=1}^{s}\{x^{k+1} \in X^{k+1} : f_{i,\,k+1}^{k+1}(x^{k+1}) \le h_i^{k+1}(\bar{x}^k, \bar{\xi}^k)\}. \tag{7}$$

Evidently, for arbitrary functions $h_i^{k+1}(\bar{x}^k, \bar{z}^k)$, $i = 1, \ldots, s$ defined on $\bar{X}^k \times \bar{Z}^k$ if $\mathcal{K}_E^{k+1}(\bar{x}^k, \bar{z}^k)$ denotes the set of efficient points of the multiobjective problem.

Find

$$\min h_i(\bar{x}^k, \bar{z}^k),\ i = 1, \ldots, s \quad \text{subject to} \quad \bar{x}^k \in \bar{X}^k,\ z^k \in Z^k, \tag{8}$$

then for $\bar{X}^k$, $\bar{Z}^k$ compact sets

$$\begin{array}{lll} \mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k,\, \bar{z}^k) & \text{nonempty for} & (\bar{x}^k,\, z^k) \in \mathcal{K}_E^{k+1}(\bar{x}^k,\, z^k) \quad \Longrightarrow \\ \mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k,\, \bar{z}^k) & \text{nonempty for} & \bar{x}^k \in \bar{X}^k,\, \bar{z}^k \in \bar{Z}^k. \end{array} \tag{9}$$

3 Some Auxiliary Assertions

According to (9), employing the proofs technique of [10] we can obtain for

$$\Lambda = \{\lambda \in R^s : \lambda = (\lambda_1, \ldots, \lambda_s), \quad \lambda_i > 0,\, i = 1, \ldots, s,\, \sum_{i=1}^{s} \lambda_i = 1\},$$

$$G^{\lambda,\, k+1}(\bar{x}^k,\, \bar{z}^k) \;= \sum_{i=1}^{s} \lambda_i h_i^{k+1}(\bar{x}^k,\, \bar{z}^k), \quad \bar{x}^k \in \bar{X}^k, \quad \bar{z}^k \in \bar{Z}^k, \quad \lambda \in \Lambda,$$

$$\mathcal{K}^{\Lambda,\, k+1}(\bar{X}^k,\, \bar{Z}^k) = \{\bar{x}^k \in \bar{X}^k,\, \bar{z}^k \in \bar{Z}^k : G^{\lambda,\, k+1}(\bar{x}^k,\, \bar{z}^k) =$$

$$\min\{G^{\lambda,\, k+1}(\bar{x}^k,\, \bar{z}^k) : \bar{x}^k \in \bar{X}^k,\, \bar{z}^k \in \bar{Z}^k\} \quad \text{for some } \lambda \in \Lambda,\, \} \tag{10}$$

the next assertion (for more details see e.g. [3], [4]).

Proposition 1. *Let $k = 0, 1, \ldots, M-1$, X^k, Z^k, be nonempty convex, compact sets. If*

1. *$\bar{\mathcal{K}}^{\Lambda,\, k+1}(\bar{X}^k,\, \bar{Z}^k)$ denotes a closure of $\mathcal{K}^{\Lambda,\, k+1}(\bar{X}^k,\, \bar{Z}^k)$,*
2. *$h_i^{k+1}(\bar{x}^k,\, \bar{z}^k)$, $i = 1, \ldots, s$, $k = 0, 1, \ldots, M-1$ are convex, continuous functions on $\bar{X}^k \times \bar{Z}^k$,*

then

$$\mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k,\, \bar{z}^k) \text{ is nonempty for every } (\bar{x}^k,\, \bar{z}^k) \in \bar{\mathcal{K}}^{\Lambda,\, k+1}(\bar{X}^k,\, \bar{Z}^k) \Longrightarrow$$

$$\mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k,\, \bar{z}^k) \text{ is nonempty for every } \bar{x}^k \in \bar{X}^k,\, \bar{z}^k \in \bar{Z}^k.$$

Proposition 2. *Let $h_i^{k+1}(\bar{x}^k,\, \bar{z}^k)$ fulfil the relation (6). If, moreover, for $i = 1, \ldots, s,\ j = 1, \ldots, k+1,\ k = 0, 1, \ldots, M-1$,*

1. *$H_i(z^k),\ f_{i,\,j}^{k+1}(x^j)$ are linear functions on $\bar{X}^M \times \bar{Z}^M$,*
2. *X^j, Z^j are polyhedral compact sets,*

then

$$\mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k,\, \bar{z}^k) \text{ is nonempty for every } (\bar{x}^k,\, \bar{z}^k) \in \mathcal{K}^{\Lambda,\, k+1}(\bar{X}^k,\, \bar{Z}^k) \Longrightarrow$$
$$\mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k,\, \bar{z}^k) \text{ is nonempty for every } \bar{x}^k \in \bar{X}^k,\, \bar{z}^k \in \bar{Z}^k.$$

Evidently, it follows from (10) and [3] that to determine, under the assumptions of Proposition 2, the set $\mathcal{K}^{\Lambda,\, k+1}(\bar{X}^k,\, \bar{Z}^k)$, a modified simplex algorithm (for parametric linear problem) can be employed. The theory of convex parametric programming can be employed whenever the assumptions of Proposition 1 are fulfilled. Furthermore, employing the results of [5] and [7] the following assertion can be proven.

Proposition 3. *Let* $k = 0,\, 1, \ldots,\, M-1$, *the assumption A.1 be fulfilled,* X^k *be nonempty, compact sets. If*

1. $\mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k,\, \bar{z}^k)$ is nonempty for every $(\bar{x}^k,\, \bar{z}^k) \in \bar{\mathcal{K}}_{E}^{\Lambda,\, k+1}(\bar{X}^k,\, \bar{Z}^k)$,

2. *for* $\bar{x}^k(i) \in \bar{X}^k,\, \bar{z}^k(i) \in \bar{Z}^k,\, i = 1, 2$ *there exists* $D > 0$ *such that*

$$\Delta[\mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k(1),\, \bar{z}^k(1)),\, \mathcal{K}_{\mathcal{F}}^{k+1}(\bar{x}^k(2),\, \bar{z}^k(2))] \leq$$
$$D\,\|h(\bar{x}^k(1),\, \bar{z}^k(1)) - h(\bar{x}^k(2),\, \bar{z}^k(2))\|, \quad h^{k+1} = (h_1^{k+1},\, \ldots,\, h_s^{k+1}),$$

3. $g_0^M(\bar{x}^M,\, \bar{z}^M)$ *is a Lipschitz function on* $\bar{X}^M \times \bar{Z}^M$,

4. *a finite* $E_{F^\varepsilon}\varepsilon\,(\varepsilon := \varepsilon^1)$ *exists,*

then $g_{\mathcal{F}}^k(\bar{x}^k,\, \bar{z}^k)$ *is a Lipschitz function on* $\bar{X}^k \times \bar{Z}^k$.

$\Delta[\cdot,\, \cdot]$ denotes the Hausdorff distance; $\|\cdot\|$ the Euclidean norm.

4 Stability and Approximation

Evidently, if we replace F^{ξ^0}, F^ε by another G^{ξ^0}, G^ε, we obtain another problem with an optimal value by $\varphi_{\mathcal{G}}(M)$. Employing the stability results (for details see e.g. [8] and [9]) we can see that if h^{k+1}, $k = 0, \ldots, M-1$ are Lipschitz in all arguments and the assumptions of Propositions 1, 2, 3 are fulfilled, then there exist $C_{W_1}^i$, $C_K^i > 0\, i = 1, \ldots s$ such that

$$|\varphi_{\mathcal{F}}(M) - \varphi_{\mathcal{G}}(M)| \leq \sum_{i=1}^{s} C_{W_1}^i \int_{R^1} |F_i^\varepsilon(z_i) - G_i^\varepsilon(z_i)|dz_i + \sum_{i=1}^{s} C_K^i |k_{F_i^\varepsilon} - k_{G_i^\varepsilon}|, \tag{11}$$

G_i^{ε}, $k_{G_i^{\varepsilon}}$, $i = 1, \ldots, s$ are one–dimensional marginal distribution functions and quantils.

The relation (11) can be employed for empirical estimates investigation and approximate solution schemes construction. However the investigation in this direction is over the possibility of this contribution.

Acknowledgement. The research was supported by the Czech Science Foundation under Grants 402/07/1113, 402/05/0115 and 402/06/1417.

References

1. Birge JR, Louveuac F (1997) Introduction to stochastic programming. Springer, Berlin
2. Dupačová J (1995) Multistage stochastic programs: the state–of–the–art and selected bibliography. Kybernetika 31:151–174
3. Ehrgott M (2005) Multicriteria optimization. Springer, Berlin 2005
4. Geoffrion JR (1968) Proper efficiency and the theory of vector Maximization. J Math Anal Appl 22:618–630
5. Kaňková V (1998) A note on multistage stochastic programming. In: Proceedings of 11th Joint Czech–Germany–Slovak Conf.: Mathematical Methods in Economy and Industry. University of Technology, Liberec (Czech Republic) 1998:45–52
6. Kaňková V (2002) A remark on the analysis of multistage stochastic programs, Markov dependence. Z angew Math Mech 82:781–793.
7. Kaňková V, Šmíd M (2004) On approximation in multistage stochastic programs: Markov dependence. Kybernetika 40:625–638
8. Kaňková V, Houda M (2006) Empirical estimates in stochastic programming. In: Proceedings of Prague Stochastics 2006 (M. Hušková and M. Janžura, eds.). MATFYZPRESS, Prague:426–436
9. Kaňková V (2007) Multistage stochastic programming problems; stability and approximation. In: Operations Research Proceedings 2006. Springer, Berlin:595–600
10. Kaňková V (2007) Stochastic Programming Problems with Recourse via Multiobjective Optimization Problems. Proceedings 15th International Scientific Conference on Mathematical Methods in Economics and Industry (K. Cechlárová, M. Halická, V. Borbeľová and V. Lacko, eds.), Univerzita P.J. Šafárika, Košice: 68–78. CD ROM 5.1 MByte 5
11. Pflug GCh (2001) Scenario tree generation for multiperiod financial optimization by optimal discretization. Math Program Ser B 89:251–271

Risk-Sensitive Average Optimality in Markov Decision Chains

Karel Sladký[1] and Raúl Montes-de-Oca[2]

[1] Institute of Information Theory and Automation
Pod Vodárenskou věží 4, 182 08 Praha 8, Czech Republic
sladky@utia.cas.cz

[2] Departamento de Matemáticas, Universidad Autónoma Metropolitana
Campus Iztapalapa, Avenido San Rafael, Atlixco # 186, Colonia Vicentina
México 09340, D.F. Mexico
momr@xanum.uam.mx

1 Introduction and Notation

We consider a Markov decision chain $X = \{X_n, n = 0, 1, \ldots\}$ with finite state space $\mathcal{I} = \{1, 2, \ldots, N\}$ and a finite set $\mathcal{A}_i = \{1, 2, \ldots, K_i\}$ of possible decisions (actions) in state $i \in \mathcal{I}$. Supposing that in state $i \in \mathcal{I}$ action $k \in \mathcal{A}_i$ is selected, then state j is reached in the next transition with a given probability p_{ij}^k and one-stage transition reward r_{ij} will be accrued to such transition.

We shall suppose that the stream of transition rewards is evaluated by an exponential utility function, say $u^\gamma(\cdot)$, with risk aversion coefficient $\gamma > 0$ (the risk averse case). Then the utility assigned to the (random) reward ξ is given by $u^\gamma(\xi) := \exp(\gamma\xi)$, and for the corresponding certainty equivalent $Z^\gamma(\xi)$ we have (E is reserved for expectation)

$$u^\gamma(Z^\gamma(\xi)) = \mathsf{E}[\exp(\gamma\xi)] \Longleftrightarrow Z^\gamma(\xi) = \gamma^{-1} \ln\{\mathsf{E}\,[\exp(\gamma\xi)]\}. \quad (1)$$

A (Markovian) policy controlling the chain, $\pi = (f^0, f^1, \ldots)$ where $f^n \in \mathcal{A} \equiv \mathcal{A}_1 \times \ldots \times \mathcal{A}_N$ for every $n = 0, 1, 2, \ldots$ and $f_i^n \in \mathcal{A}_i$ is the decision at the nth transition when the chain X is in state i. A policy which takes at all times the same decision rule, i.e. $\pi \sim (f)$, is called stationary.

Let $\xi_n = \sum_{k=0}^{n-1} r_{X_k, X_{k+1}}$ be the stream of transition rewards received in the n next transitions of the considered Markov chain X, and similarly let $\xi^{(m,n)}$ be reserved for the total (random) reward obtained from the mth up to the nth transition (obviously, $\xi_n = r_{X_0, X_1} + \xi^{(1,n)}$).

Supposing that the chain starts in state $X_0 = i$ and policy $\pi = (f^n)$ is followed, then for expected utility in the n next transitions, the corresponding certainty equivalent, and for mean value of the certainty equivalent we have (E_i^π denotes expectation if policy π is followed and $X_0 = i$)

$$U_i^\pi(\gamma, 0, n) := \mathsf{E}_i^\pi[\exp(\gamma \sum_{k=0}^{n-1} r_{X_k, X_{k+1}})] \tag{2}$$

$$Z_i^\pi(\gamma, 0, n) := \gamma^{-1} \ln\{\mathsf{E}_i^\pi[\exp(\gamma \sum_{k=0}^{n-1} r_{X_k, X_{k+1}})]\}. \tag{3}$$

$$J_i^\pi(\gamma, 0) := \limsup_{n\to\infty} n^{-1} Z_i^\pi(\gamma, 0, n). \tag{4}$$

In what follows we shall often abbreviate $U_i^\pi(\gamma, 0, n)$, $Z_i^\pi(\gamma, 0, n)$ and $J_i^\pi(\gamma, 0)$ respectively by $U_i^\pi(\gamma, n)$, $Z_i^\pi(\gamma, n)$ and $J_i^\pi(\gamma)$ respectively. Similarly $\boldsymbol{U}^\pi(\gamma, n)$ is reserved for the (column) vector whose ith element equals $U_i^\pi(\gamma, n)$. The symbol $\boldsymbol{e}$ is a unit (column) vector. $\boldsymbol{Q}(f)$ is an $N \times N$ nonnegative matrix with elements $q_{ij}(f_i) := p_{ij}^{f_i} \cdot \mathrm{e}^{\gamma r_{ij}}$.

In this note we focus attention on the asymptotic behavior of the expected utility and the corresponding certainty equivalents, similarly as in [2, 4]. However, our analysis is based on the properties of a collection of nonnegative matrices arising in the recursive formulas for the growth of expected utilities is not restricted to irreducible matrices.

2 Risk-Sensitive Optimality and Nonnegative Matrices

Conditioning in (2) on X_1 (since $u^\gamma(\xi_n) = \mathsf{E}[u^\gamma(r_{X_0,X_1}) \cdot u^\gamma(\xi^{(1,n)})|X_1 = j]$) from (2) we immediately get

$$U_i^\pi(\gamma, 0, n) = \sum_{j\in\mathcal{I}} q_{ij}^{f_i^0} \cdot U_j^\pi(\gamma, 1, n) \qquad \text{with} \quad U_i^\pi(\gamma, n, n) = 1 \tag{5}$$

or in vector notation

$$\boldsymbol{U}^\pi(\gamma, 0, n) = \boldsymbol{Q}(f^0) \cdot \boldsymbol{U}^\pi(\gamma, 1, n) \qquad \text{with} \quad \boldsymbol{U}^\pi(\gamma, n, n) = \boldsymbol{e}. \tag{6}$$

Iterating (6) we get if policy $\pi = (f^n)$ is followed

$$\boldsymbol{U}^\pi(\gamma, n) = \boldsymbol{Q}(f^0) \cdot \boldsymbol{Q}(f^1) \cdot \ldots \cdot \boldsymbol{Q}(f^{n-1}) \cdot \boldsymbol{e}. \tag{7}$$

To study the properties of (7) we shall employ following facts:

(i) (See e.g. [1, 3].) If every $\boldsymbol{Q}(f)$ with $f \in \mathcal{A}$ is irreducible, then there exists $f^* \in \mathcal{A}$, and $\boldsymbol{v}(f^*) > \boldsymbol{0}$ such that for any $f \in \mathcal{A}$

$$\boldsymbol{Q}(f) \cdot \boldsymbol{v}(f^*) \le \boldsymbol{Q}(f^*) \cdot \boldsymbol{v}(f^*) = \rho(f)\boldsymbol{v}(f^*). \tag{8}$$

Moreover, (8) can be fulfilled even for reducible matrices, on condition that $\boldsymbol{Q}(f^*)$ can be decomposed as

$$\boldsymbol{Q}(f^*) = \begin{bmatrix} \boldsymbol{Q}_{(\mathrm{NN})}(f^*) & \boldsymbol{Q}_{(\mathrm{NB})}(f^*) \\ \mathbf{0} & \boldsymbol{Q}_{(\mathrm{BB})}(f^*) \end{bmatrix} \tag{9}$$

where the spectral radius of $\boldsymbol{Q}_{(\mathrm{BB})}(f^*)$ is equal to $\rho(f^*)$ and the spectral radius of (possibly reducible) $\boldsymbol{Q}_{(\mathrm{NN})}(f^*)$ is less than $\rho(f^*)$. (Observe that (9) well corresponds to the canonical decomposition of transition probability matrix with r classes of recurrent states.)

(ii) (See [6, 7, 8].) There exists suitable labelling of states such that: Every $\boldsymbol{Q}(f)$ with $f \in \mathcal{A}$ is block triangular, i.e.

$$\boldsymbol{Q}(f) = \begin{bmatrix} \boldsymbol{Q}_{11}(f) & \boldsymbol{Q}_{12}(f) & \dots & \boldsymbol{Q}_{1s}(f) \\ \mathbf{0} & \boldsymbol{Q}_{22}(f) & \dots & \boldsymbol{Q}_{2s}(f) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \boldsymbol{Q}_{ss}(f) \end{bmatrix} \tag{10}$$

where all $\boldsymbol{Q}_{ii}(f)$ have fixed dimensions, and are the "biggest" submatrices of $\boldsymbol{Q}(f)$ having strictly positive right eigenvectors corresponding to the maximum possible spectral radii of the corresponding submatrices, i.e. there exists $\boldsymbol{Q}(f^*)$ along with $\boldsymbol{v}_i(f^*) > \mathbf{0}$ (i. e. strictly positive) such that for all $i = 1, 2, \dots, s$

$$\rho_i(f^*) \geq \rho_i(f); \quad \rho_i(f^*) \geq \rho_{i+1}(f^*) \tag{11}$$

$$\boldsymbol{Q}_{ii}(f) \cdot \boldsymbol{v}_i(f^*) \leq \boldsymbol{Q}_{ii}(f^*) \cdot \boldsymbol{v}_i(f^*) = \rho_i(f^*)\boldsymbol{v}_i(f^*) \tag{12}$$

Observe that $\rho_1(f^*) = \rho(f^*)$ and that each diagonal block $\boldsymbol{Q}_{ii}(f)$ in (10) may be reducible, and if $\boldsymbol{Q}_{ii}(f^*)$ is reducible then it can a decomposed according to (9).

Throughout this note we make the following assumption:

Assumption GA. A strict inequalities hold in the second part of (11), i.e.:

$$\rho_1(f^*) > \rho_2(f^*) > \ldots > \rho_s(f^*). \tag{13}$$

Obviously, since eigenvectors are unique up to a multiplicative constant, on condition that $\boldsymbol{v}(f^*) > \mathbf{0}$ we can choose $\boldsymbol{v}(f^*) \geq \boldsymbol{e}$ and $\boldsymbol{v}(f^*) \leq \boldsymbol{e}$ respectively, and on replacing in (7) $\boldsymbol{e}$ by $\boldsymbol{v}(f^*)$ recursion (7) will immediately yields upper and lower bounds on $\boldsymbol{U}^{\pi}(\gamma, n)$ respectively.

In particular, for an easy case with $\boldsymbol{Q}(f)$'s irreducible, or if at least condition (8) holds even for some reducible matrix $\boldsymbol{Q}(f^*)$, we can immediately conclude that for $\boldsymbol{v}(f^*) \geq \boldsymbol{e}$ and any policy $\pi = (f^k)$

$$\prod_{k=0}^{n-1} \boldsymbol{Q}(f^k) \cdot \boldsymbol{v}(f^*) \leq (\boldsymbol{Q}(f^*))^n \cdot \boldsymbol{v}(f^*) = (\rho^*)^n \cdot \boldsymbol{v}(f^*) \tag{14}$$

and hence the asymptotic behaviour of $\boldsymbol{U}^\pi(\gamma, n)$ (or of $\boldsymbol{U}^\pi(\gamma, m, n)$ if m is fixed) heavily depends of on $\rho(f^*) \equiv \rho^*$, and elements of $\prod_{k=0}^{n-1} \boldsymbol{Q}(f^k) \cdot \boldsymbol{v}(f^*)$ must be bounded from above by $(\rho(f^*))^n \cdot \boldsymbol{v}(f^*)$.

Similarly, on selecting $\boldsymbol{v}(f^*) \leq \boldsymbol{e}$, we get for stationary policy $\pi^* \sim (f^*)$:

$$(\boldsymbol{Q}(f^*))^n \cdot \boldsymbol{e} \geq (\rho^*)^n \, \boldsymbol{v}(f^*). \tag{15}$$

Hence the growth of $\boldsymbol{U}^{\pi^*}(\gamma, n)$ is also bounded from below by $(\rho(f^*))^n \cdot \boldsymbol{v}(f^*)$.

Now (cf. (10)–(12)) let us consider the reducible case. Obviously, if we choose for all i $\boldsymbol{v}_i(f^*) \leq \boldsymbol{e}$ and ignore in $\boldsymbol{Q}(f^*)$ all its off-diagonal blocks, we can easily see that for $\pi^* \sim (f^*)$ the growth of each $\boldsymbol{U}_i^\pi(\gamma, n)$ is non-smaller than $(\rho_i(f^*))^n$. Moreover, we have already shown that the maximal growth of $\boldsymbol{U}_s^\pi(\gamma, n)$ is governed by $\rho_s(f^*)$ and is attained for stationary policy $\pi^* \sim (f^*)$. Hence it suffices to show by induction on $i = s-1, \ldots, 1$ that the growth of each $\boldsymbol{U}_i^\pi(\gamma, n)$ is also dominated by the appropriate powers of $\rho_i(f^*)$.

We present only a sketch of the proof for $s = 2$, i.e., when each $\boldsymbol{Q}(f)$ can be decomposed as

$$\boldsymbol{Q}(f) = \begin{bmatrix} \boldsymbol{Q}_{11}(f) & \boldsymbol{Q}_{12}(f) \\ \boldsymbol{0} & \boldsymbol{Q}_{22}(f) \end{bmatrix} \tag{16}$$

with $\rho_1(f^*) > \rho_2(f^*)$ and $\varepsilon^* := \rho_2(f^*)/\rho_1(f^*) < 1$. Then by (12) we have $\boldsymbol{Q}_{ii}(f) \cdot \boldsymbol{v}_i(f^*) \leq \boldsymbol{Q}_{ii}(f^*) \cdot \boldsymbol{v}_i(f^*) = \rho_i(f^*)\boldsymbol{v}_i(f^*)$, for $i = 1, 2$, and since

$$(\boldsymbol{Q}(f))^n = \begin{bmatrix} (\boldsymbol{Q}_{11}(f))^n & \sum\limits_{k+\ell=n-1} (\boldsymbol{Q}_{11}(f))^k \boldsymbol{Q}_{12}(f) (\boldsymbol{Q}_{22}(f))^\ell \\ \boldsymbol{0} & (\boldsymbol{Q}_{22}(f))^n \end{bmatrix} \tag{17}$$

we can conclude that if $\boldsymbol{v}_1(f) \geq \boldsymbol{e}$, $\boldsymbol{v}_2(f) \geq \boldsymbol{e}$ we have for any policy $\pi = (f^n)$

$$\begin{bmatrix} \boldsymbol{U}_1^\pi(\gamma, n) \\ \boldsymbol{U}_2^\pi(\gamma, n) \end{bmatrix} \leq \begin{bmatrix} (\rho_1(f^*))^{n-1} \left\{ \rho_1(f^*) + \alpha \frac{1}{1-\varepsilon^*} \cdot \right\} \cdot \boldsymbol{v}_1(f^*) \\ (\rho_2(f^*))^n \cdot \boldsymbol{v}_2(f^*) \end{bmatrix}$$

and the growth of $\boldsymbol{U}_1^\pi(\gamma, n)$ is dominated by $\rho_1(f^*)$.

In general, the growth of $\boldsymbol{U}_i^\pi(\gamma, n)$ is dominated by $\rho_i(f^*)$ that can be obtained along with $\boldsymbol{v}_i(f^*) > \boldsymbol{0}$ (unique up to a multiplicative constant) as a solution of (12). Denoting elements of $\boldsymbol{v}_i(f) > \boldsymbol{0}$ by $v_{(i),j}(f)$

(for $j = 1, \ldots, N_i$) and elements of $N_i \times N_i$ matrix $\boldsymbol{Q}_{ii}(f)$ by $q_{(i),jk}(f)$ (recall that $q_{(i),jk}(f) = p^f_{(i),jk} \cdot \mathrm{e}^{\gamma r_{ij}}$) from (12) we get for $g_{(i)}(f)$, $w_{(i),j}(f)$ $(j = 1, \ldots, N_i)$ defined by $v_{(i),j}(f) = \mathrm{e}^{\gamma w_{(i),j}(f)}$, $\rho_i(f) = \mathrm{e}^{\gamma g_{(i)}(f)}$ the following set of equations for $\ell = 1, \ldots, N_i$

$$\mathrm{e}^{\gamma(g_{(i)}(f)+w_{(i),\ell}(f))} = \max_{f\in\mathcal{A}}\{\sum_{j\in\mathcal{I}_{(i)}} p^f_{(i),\ell j} \cdot \mathrm{e}^{\gamma(r_{(i),\ell j}+w_{(i),j}(f))}\} \tag{18}$$

called γ-*average reward optimality equation.*
In the multiplicative form (used before) we write for $\ell = 1, \ldots, N_i$

$$\rho_i(f)\, v_{(i),\ell}(f) = \max_{f\in\mathcal{A}}\{\sum_{j\in\mathcal{I}_{(i)}} p^f_{(i),\ell j} \cdot \mathrm{e}^{\gamma r_{(i),\ell j}} \cdot v_{(i),j}(f)\}. \tag{19}$$

Observe that the solution to (18), resp. (19), is unique up to an additive constant, resp. multiplicative constant.

3 Finding Optimal Solutions by Value Iterations

Consider the following dynamic programming recursion for $n = 0, 1, \ldots,$

$$\widehat{\boldsymbol{U}}(n+1) = \max_{f\in\mathcal{A}} \boldsymbol{Q}(f)\cdot\widehat{\boldsymbol{U}}(n) = \boldsymbol{Q}(\hat{f}^{(n)})\cdot\widehat{\boldsymbol{U}}(n) \quad \text{with} \quad \widehat{\boldsymbol{U}}(0) = \boldsymbol{e}. \tag{20}$$

If there exists ρ^* and $\boldsymbol{v}^* > \boldsymbol{0}$ such that

$$\rho^*\boldsymbol{v}^* = \max_{f\in\mathcal{A}} \boldsymbol{Q}(f)\cdot\boldsymbol{v}^* = \boldsymbol{Q}(f^*)\cdot\boldsymbol{v}^* \tag{21}$$

with $\boldsymbol{Q}(f^*)$ aperiodic then (see [5], Theorem 3.4)

i) $\widehat{\boldsymbol{U}}(n) \to \boldsymbol{v}^*$ as $n \to \infty$;

ii) Let $\rho_{\max}(n) := \max_{i\in\mathcal{I}} \hat{U}_i(n+1)/\hat{U}_i(n)$, $\rho_{\min}(n) := \min_{i\in\mathcal{I}} \hat{U}_i(n+1)/\hat{U}_i(n)$, then the sequence $\{\rho_{\max}(n)\}$ is nonincreasing, $\{\rho_{\min}(n)\}$ is nondecreasing, and

$$\lim_{n\to\infty} \rho_{\max}(n) = \lim_{n\to\infty} \rho_{\min}(n) = \rho^*. \tag{22}$$

Hence for the corresponding values of certainty equivalents we get

$$Z_i^{\pi^*}(\gamma, n) = \gamma^{-1}\cdot\ln[U_i^{\pi^*}(\gamma, n)] = \gamma^{-1}\cdot[n\ln(\rho^*) + w_i] \tag{23}$$

and for the mean value of certainty equivalents we have

$$J_i^{\pi^*}(\gamma) = \gamma^{-1}\cdot\ln[\rho^*] \qquad \text{for} \quad i = 1, 2, \ldots, N. \tag{24}$$

The above procedures also enables to generate upper and lower bounds on the mean value of certainty equivalents. In particular, for

$$J_{\max}(\gamma, n) := \gamma^{-1}\ln[\rho_{\max}(n)], \quad J_{\min}(\gamma, n) := \gamma^{-1}\ln[\rho_{\min}(n)]$$

the sequence $\{J_{\max}(\gamma, n), n = 0, 1, \ldots\}$, resp. $\{J_{\min}(\gamma, n), n = 0, 1, \ldots\}$,

is nonincreasing, resp. nondecreasing, and

$$\lim_{n\to\infty} J_{\max}(\gamma,n) = \lim_{n\to\infty} J_{\min}(\gamma,n) = J_i^{\pi^*} \quad \text{independently of} \quad i \in \mathcal{I}.$$

If there exists no $\boldsymbol{v}^* > \mathbf{0}$ such that (21) holds we can proceed as follows: Suppose (for simplicity) that $\{\boldsymbol{Q}(f), f \in \mathcal{A}\}$ can be decomposed into block-triangular form with two diagonal blocks $\boldsymbol{Q}_{11}(f)$, $\boldsymbol{Q}_{22}(f)$ (i.e. $\boldsymbol{Q}_{21}(f) = \mathbf{0}$ for any $f \in \mathcal{A}$), and there exists $f^* \in \mathcal{A}$ and $\boldsymbol{v}_1(f^*) > \mathbf{0}$, $\boldsymbol{v}_2(f^*) > \mathbf{0}$ such that $\rho_1(f^*) > \rho_2(f^*)$, and for any $\boldsymbol{Q}(f)$ with $f \in \mathcal{A}$, for $i = 1, 2$

$$\boldsymbol{Q}_{ii}(f) \cdot \boldsymbol{v}_i(f^*) \le \rho_i(f^*)\, \boldsymbol{v}_i(f^*) = \boldsymbol{Q}_{ii}(f^*) \cdot \boldsymbol{v}_i(f^*). \tag{25}$$

Let the rows of $\boldsymbol{Q}_{11}(f)$, resp. $\boldsymbol{Q}_{22}(f)$, be labelled by numerals from $\mathcal{I}_1$, resp. $\mathcal{I}_2$. (Obviously, $\mathcal{I} = \mathcal{I}_1 \cup \mathcal{I}_2$.) Then on iterating (25) we get:

$$\lim_{n\to\infty} \hat{U}_i(n+1)/\hat{U}_i(n) = \rho_1(f^*), \quad \text{for any} \quad i \in \mathcal{I}_1$$
$$\lim_{n\to\infty} \hat{U}_i(n+1)/\hat{U}_i(n) = \rho_2(f^*), \quad \text{for any} \quad i \in \mathcal{I}_2, \quad \text{and for}$$
$$\rho^{(1)}_{\max}(n) := \max_{i\in\mathcal{I}_1} \hat{U}_i(n+1)/\hat{U}_i(n), \qquad \rho^{(1)}_{\min}(n) := \min_{i\in\mathcal{I}_1} \hat{U}_i(n+1)/\hat{U}_i(n)$$

we can conclude that $\{\rho^{(1)}_{\max}(n)\}$ nonincreasing, $\{\rho^{(1)}_{\min}(n)\}$ nondecreasing, and $\lim_{n\to\infty} \rho^{(1)}_{\max}(n) = \lim_{n\to\infty} \rho^{(1)}_{\min}(n) = \rho_1(f^*)$, where $\rho_1(f^*)$ is the maximum possible growth rate that can occur in states from $\mathcal{I}_1$. The same holds also for mean values of the corresponding certainty equivalents, and also for states from $\mathcal{I}_2$.

Acknowledgement. The research of Karel Sladký was supported by the Czech Science Foundation under Grants 402/05/0115 and 402/04/1294.

References

1. Berman A and Plemmons RJ (1979) Nonnegative Matrices in the Mathematical Sciences. Academic Press, New York
2. Cavazos-Cadena R, Montes-de-Oca R (2003) The value iteration algorithm in risk-sensitive average Markov decision chains with finite state space. Math Oper Res 28:752–756
3. Gantmakher FR (1959) The Theory of Matrices. Chelsea, London
4. Howard RA, Matheson J (1972) Risk-sensitive Markov decision processes. Manag Sci 23:356–369
5. Sladký K (1976) On dynamic programming recursions for multiplicative Markov decision chains. Math Programming Study 6:216–226
6. Sladký K (1980) Bounds on discrete dynamic programming recursions I. Kybernetika 16:526–547
7. Whittle P (1983) Optimization Over Time – Dynamic Programming and Stochastic Control. Volume II, Chapter 35, Wiley, Chichester
8. Zijm WHM (1983) Nonnegative Matrices in Dynamic Programming. Mathematical Centre Tract, Amsterdam

A Stochastic Programming Model with Decision Dependent Uncertainty Realizations for Technology Portfolio Management

Senay Solak[1], John-Paul Clarke[2], Ellis Johnson[1], and Earl Barnes[1]

[1] School of Industrial and Systems Engineering, Georgia Institute of Technology
[2] School of Aerospace Engineering, Georgia Institute of Technology

1 Introduction

Technology development involves research and development (R&D) projects aimed to design, test and improve a technology, or the process of building a technology. Technology development is often an essential part of the operational strategy of an organization, during which deployment or implementation decisions are made. In most cases, organizations have several potential technologies with different characteristics that they can choose to invest in and develop using available resources. Selection of projects and allocation of the resources to the selected projects are important decisions with huge economic implications for an organization.

Despite the importance and economic significance of R&D project portfolio selection and the existence of several operations research models, the industrial use of these models has been limited. This is mainly due to the fact that none of the proposed models has been able to capture the full range of complexity that exists in technology development portfolios. The proposed models include capital budgeting models, which capture interdependencies between different projects, but fail to model the uncertainty in returns and required investments[3]. More recent project portfolio models capture both the uncertainty in returns and interdependencies. However, these models assume that the required cash flows for projects are known, and the investment decisions consist of binary starting or stopping decisions for projects [1, 2]. In addition to these models, most strategic planners and technology portfolio managers rely on tools based on expert opinions. Clearly, these tools are very limited in their ability to fully quantify the complicated return

and investment structure inherent in technology portfolios. Hence, it is essential that advanced decision tools to determine optimal technology portfolios are developed. This study fills this gap by developing a detailed model and practical solution techniques for the technology portfolio management problem.

2 Mathematical Representation and Model

Assume a set $\mathcal{N}$ of technologies with annual performance levels $Z_i \in \mathbf{R}^+$, implementation times $\Delta_i \in \mathbf{R}^+$, required investment levels $\theta_i \in \mathbf{R}^+$, annual fixed activity costs $f_i \in \mathbf{R}^+$ and a set of depending technologies $\mathcal{D}_i \subset \mathcal{N}$, for each $i \in \mathcal{N}$. Although only two-way dependencies between technologies are used in this study, the proposed models can be extended to handle multi-way dependencies in a similar fashion. We let $Z_{ij} \in \mathbf{R}$ be the joint annual performance level for technology $i \in \mathcal{N}$ and $j \in \mathcal{D}_i$, and define it as a function of Z_i and Z_j. Furthermore, a sequence of investment planning periods $t = 1, 2, ..., T$ with available resource levels, i.e. budgets $B_t \in \mathbf{R}^+$, are assumed. The objective is to determine an investment schedule such that some function of the total discounted return over an infinite time horizon is maximized while total investment in a given period t does not exceed B_t. In typical applications, the decision maker is interested in the investment schedule for the current period only, which should take into account future realizations of the parameters. Hence, a realistic assumption is that the problem will be solved each planning period to determine the best investment policy for that period, considering the past and future investments.

In practice, almost all of the above parameters may contain a certain level of uncertainty. However, observational analyses suggest that the level of variance is significant only in two of the parameters, namely the returns Z_i and required investment levels θ_i . Note that Z_{ij} is defined as a function of Z_i and Z_j. Hence, we approximate all other parameters with their expected values, and assume that joint and marginal probability distributions of the returns and required investment levels for the technologies are known or well estimated.

The decision process in the technology portfolio management problem consists of recourse actions, by which the portfolio can be rebalanced at each period. Hence, an appropriate approach is to formulate the problem as a recourse problem, in which recourse actions can be taken after uncertainty is disclosed over the investment period. This decision process can be described as follows.

The resource requirement θ_i for each technology i is known with certainty at the end of period t^i_θ, in which total investment in the technology exceeds a threshold level Θ^θ_i, i.e. $t^i_\theta = \min_t\{t|\sum_{t'\leq t} x_{it'} \geq \Theta^\theta_i\}$, where x_{it} represents the investment for technology i in period t. Similarly, we assume that the uncertainty in the return of a technology is revealed gradually over its development based on certain threshold levels. This process is modeled by assuming that an initial performance assessment $\hat{Z}_i$ will be available at the end of period $t^i_z = \min_t\{t|\sum_{t'\leq t} x_{it'} \geq \Theta^z_i\}$ upon investing an amount of Θ^z_i in the technology. As a result of this assessment, probabilities of different performance levels are updated. This assumption enables the modeling of the option of terminating a project if the initial assessment suggests that the probability of a high return is low for the technology. Gradual resolution of uncertainty can be explained further as follows. Assume that Z_i can be realized at one of two levels: L, H with pre-development probabilities p_L and p_H, respectively. After investing an amount Θ^z_i in this technology, an estimate $\hat{Z}_i$ is made, which can be seen as an intermediate realization of the uncertain parameter. If all uncertainty is resolved when technology development is over, then the probabilities for the actual realization of the possible outcomes can depend on the intermediate realization. If the development phase is continued, return Z_i will be known with certainty once all of the required resources are invested in technology i.

The described process can be modeled as a multistage stochastic program, in which the uncertainty is in required investment levels, updated return estimates and final return levels. As in many other stochastic programs, it is reasonable to assume for the technology portfolio management problem that the random vector ξ has finite support or has a discrete distribution with K possible realizations, i.e. scenarios, $\xi^k := (\theta^k_i, \hat{Z}^k_i, Z^k_i),\ k = 1, \ldots, K$ with corresponding probabilities p_k. Then, it becomes possible to express the problem as one large mathematical program. The details of the resulting multistage stochastic programming model are provided in [4].

3 An Efficient Solution Procedure

The formulation in [4] is significant, as it is directly amenable to scenario decomposition, unlike the previous models suggested for stochastic programming problems with endogenous uncertainty. However, a sampling procedure is also necessary due to the large number of scenarios for realistic instances of the problem. Hence, we use the sample

average approximation (SAA) method to handle large instances of the problem. Let $\xi^1, ..., \xi^N$ be an i.i.d. random sample of N realizations of the random vector ξ. Then the SAA problem is:

$$\max_{\mathbf{x}\in\mathcal{X}}\{\hat{g}_N(\mathbf{x}) = \frac{1}{N}\sum_{l=1}^{N} G(\mathbf{x}, \xi^l)\} \tag{1}$$

Since the computational complexity of the SAA problem increases exponentially with the value of N, it is more efficient to select a smaller sample size N, and solve several SAA problems with i.i.d. samples. However, effective implementation of the above sampling procedure requires that the SAA problems can be solved efficiently for relatively large values of the sample size N. As an efficient solution procedure for the SAA problem, we propose a Lagrangian relaxation and decomposition scheme coupled with a lower bounding heuristic, which we name as the feasible dual conversion algorithm. The development of such a procedure is important, since for most multistage stochastic problems, even finding a feasible solution to serve as a lower bound is difficult. This general solution algorithm for the technology portfolio management problem can be summarized as follows:

Step 1. Obtain N samples from the set of scenarios, and form the SAA problem with these scenarios.

Step 2. Perform Lagrangian relaxation on the SAA problem, decomposing the problem into individual scenario subproblems.

Step 3. Use subgradient algorithm with the proposed step size measure to obtain an upper bound for the SAA problem.

3a. If computationally feasible, solve the LP relaxation of the deterministic equivalent of the multistage model, and set the corresponding dual values as the initial Lagrangian multipliers. Use a rounding heuristic to obtain an initial lowerbound on the problem.

3b. At each iteration j of the algorithm, determine a lowerbound for the scenario subproblems by calculating $\dot{L}_l(\dot{\mathbf{x}}^l, \lambda^{j+1}, \mu^{j+1})$, and selecting the minimum.

3c. At every f_o iterations, apply the feasible dual conversion algorithm, to obtain a lowerbound for the SAA problem, as well as for the scenario subproblems.

3d. Use the best lowerbounds for the scenario subproblems as the starting solution for the subproblems at iteration $j + 1$.

4. Calculate the duality gap upon convergence of the subgradient algorithm. If the gap is less than or equal to ϵ, go to step 5. Else, use branch and bound to close the duality gap, by branching on the nonanticipativity conditions.

5. Repeat Steps 1-4 M times. Each solution is a candidate solution for the true problem.

6. For some or all of the candidate solutions, perform N' replications by fixing the values of the first stage variables according to the solution, and repeating steps $1-4$ with the fixed values to estimate the objective value of the candidate solutions.

7. Select a solution as the best solution using an appropriate criterion.

3.1 The Feasible Dual Conversion Algorithm

The objective function of the technology portfolio management problem is defined by the values of the binary variables β_{it}, which represent the periods that the return realizations begin. Hence, the corresponding values in a given Lagrangian dual solution describe some infeasible investment policy in which nonanticipativity constraints are not enforced but are only penalized. Clearly, the optimal objective value of the primal problem is expected to be as close as possible or comparable to that of this infeasible policy. Thus, one can obtain a "good" investment policy by converting the dual solution into a feasible solution by a minimal change in the β_{it} values in the Lagrangian dual solution. We present below an algorithm to achieve this. The feasible dual conversion algorithm performs such conversions in a systematic way that ensures the quality of the resulting solution as well as computational efficiency. The steps of the algorithm are as follows:

Step 1. *Initialization* : Let β^j represent the vector of corresponding values in a solution to the Lagrangian dual problem at iteration j of the subgradient algorithm for dual variables λ^j and μ^j. Let $\underline{\beta}^l_{it}$, $\underline{\hat{g}}_N$, $\underline{L}_l$ be the lowerbounds on β^l_{it}, $\hat{g}_N$ and L_l. Choose a scenario subset size S. Set $\underline{\beta}^l_{it} = 0$ for all i, t, l, $\mathbf{S} = \emptyset$, $\mathbf{S}' = \emptyset$, $\mathbf{N} = \{l_1, l_2, \ldots, l_N\}$.

Step 2. *Scenario subset selection* : Rank all $s \in \mathbf{N}$ according to scenario objectives L^j_s, and form subset $\mathbf{S}$ by selecting the first S scenarios among the ranked scenarios in $\mathbf{N}$. Let $\mathbf{S}' = \mathbf{S}' \cup \mathbf{S}$ and $\mathbf{N} = \mathbf{N} \backslash \mathbf{S}$.

Step 3. *Variable fixing* : For each $s \in \mathbf{S}$, determine t^s_o in which s becomes distinguishable from all other scenarios according to β^s_{it}, i.e.

$$t^s_o = \min_t \{t | \min_{s' \neq s} \{ \sum_{j \in Y_{ss'}} (\beta^s_{j,t+\Delta_j} + \beta^{s'}_{j,t+\Delta_j}) + \sum_{j \in H_{ss'}} (\beta^s_{j,t+\Delta_j} + \beta^{s'}_{j,t+\Delta_j}) \} \geq 1\} \tag{2}$$

For each $i \in \mathcal{N}$ such that $\beta^s_{i,t+\Delta_i} = 1$, and $t \leq t^s_o$; if $\beta^s_{i,t+\Delta_i} - \beta^s_{i,t+\Delta_i - 1} = 1$, then set $\underline{\beta}^s_{i,t+\Delta_i} = 1$.

Step 4. *Feasibility determination*: Check feasibility with the lower bounds on β_{it}^s for the scenario set $\mathbf{S}'$. If feasible, let $\dot{\beta}_{it}^s$ represent the corresponding values in this solution, and fix $\beta_{it}^s = \dot{\beta}_{it}^s$. If $\mathbf{N} \neq \emptyset$, go to Step 2.

Step 5. *Minimum dual conversion* : If infeasible, determine the minimum number of relaxations r_o required on $\underline{\beta}_{it}^s = 1$ for $s \in \mathbf{S}$ to obtain a feasible solution. Find the best possible feasible solution that can be achieved by relaxing at most r_o of the bounds $\underline{\beta}_{it}^s$. Fix $\beta_{it}^s = \dot{\beta}_{it}^s$. If $\mathbf{N} \neq \emptyset$, go to Step 2.

Step 6. *Bound calculation* : Let $\dot{\mathbf{x}}$ and $\dot{g}_N$ represent the final solution vector and objective function value. If $\dot{g}_N > \underline{\hat{g}}_N$, set $\underline{\hat{g}}_N = \dot{g}_N$. For each scenario l, calculate $\dot{L}_l(\dot{\mathbf{x}}, \lambda^{j+1}, \mu^{j+1})$. If $\dot{L}_l > \underline{L}_l^{j+1}$, set $\underline{L}_l^{j+1} = \dot{L}_l$.

4 Conclusions

The technology portfolio optimization problem is a difficult practical problem, for which a comprehensive model and solution methodology has not been developed in several limited approaches in the literature. In this study, we fill this gap by formally defining and effectively modeling several complexities that are inherent in this problem, and developing an efficient solution procedure. Implementation of the proposed models in project portfolio selection by organizations will lead to significant increases in returns, as all relevant inputs and uncertainty are captured in the models, as opposed to existing project portfolio selection tools.

References

1. F. Ghasemzadeh, N. Archer, and P. Iyogun. A zero-one model for project portfolio selection and scheduling. *The Journal of the Operational Research Society*, 50:745–755, 1999.
2. J. Gustafsson and A. Salo. Contingent portfolio programming for the management of risky projects. *Operations Research*, 53(6):946–956, 2005.
3. D. G. Luenberger. *Investment Science*. Oxford Univ. Press, New York, 1998.
4. S. Solak, J. Clarke, E. Johnson, E. Barnes, and P. Collopy. Optimization of technology development portfolios. *working paper*, 2007.

Part IV

Artificial Intelligence, Business Intelligence and Decision Support

A Neural Network Based Decision Support System for Real-Time Scheduling of Flexible Manufacturing Systems

Derya Eren Akyol and Ozlem Uzun Araz

Department of Industrial Engineering, Dokuz Eylul University
35100 Bornova-Izmir, Turkey
derya.eren@deu.edu.tr, ozlem.uzun@deu.edu.tr

Summary. The objective of this study is to develop a neural network based decision support system for selection of appropriate dispatching rules for a real-time manufacturing system, in order to obtain the desired performance measures given by a user, at different scheduling periods. A simulation experiment is integrated with a neural network to obtain the multi-objective scheduler, where simulation is used to provide the training data. The proposed methodology is illustrated on a flexible manufacturing system (FMS) which consists of several number of machines and jobs, loading/unloading stations and automated guided vehicles (AGVs) to transport jobs from one location to another.

1 Introduction

Scheduling as being part of production planning and control, plays an important role in the whole manufacturing process. Although scheduling is a well researched area, classical scheduling theory has been little used in real manufacturing environments due to the assumption that the scheduling environment is static. In a static scheduling environment where the system attributes are deterministic, different analytical tools such as mathematical modeling, dynamic programming, branch-and-bound methods can be employed to obtain the optimal schedule. However, scheduling environment is usually dynamic in real world manufacturing systems and the schedule developed beforehand may become inefficient in a dynamically changing and uncertain environment. One of the most applied solutions to the dynamic scheduling problems is the use of dispatching rules. Over the years, many dispatching rules have been studied by many researchers [4, 5, 11, 12]. However, the choice of

a dispatching rule depends on the performance criteria considered, and the system configuration and conditions, in other words, on the state of the system and no single rule has been found to perform well for all the performance criteria and all possible states of the system. Therefore, a flexible scheduling method that can handle system variation which results from the change of manufacturing conditions is needed to select the best dispatching rule for each particular state of the system.

Having the ability to learn and generalize for new cases in short time, in recent years, artificial neural networks (ANNs) have provided a means of tackling dynamic scheduling problems. A number of different ANN approaches have been developed for the solution of dynamic scheduling problems, most of which are based on the use of backpropagation networks [2, 3, 6, 7, 10]. However, the use of competitive networks in dynamic scheduling environments is sparse. Min et al. [8] designed a dynamic and real time FMS scheduler by combining the competitive neural network and search algorithm to meet the multiple objectives given by the FMS operator. Min and Yih [9] integrated simulation and a competitive neural network and develop a multi-objective scheduler to select dispatching rules for both machine and vehicle initiated dispatching decision variables (for a detailed survey, see [1]).

In this paper, we introduce a multi-objective scheduler based on the integration of simulation and a competitive network, parallel to the work of Min and Yih [9]. Here, we study a Flexible Manufacturing System (FMS), where the user considers to improve the value of only one performance measure at each interval. However, by giving the possibility to consider different performance measures at different intervals, the proposed scheduler serves as a multi-objective tool.

2 Proposed Scheduler

The proposed scheduler is the combination of a simulation model and a neural network. The data needed to train the proposed network is generated through simulation using Arena software (version 10). Two different 5000 minute scheduling intervals are taken into consideration during the data collection step. 250 alternative scenarios implementing different dispatching rules in each interval are used as training set for the proposed network. Each scenario including two different sets of decision variables was simulated for 5 replications and the performance measures and the values of the system status variables at the end of each interval are obtained as simulation outputs. Two sets of decision rules, together with the simulation outputs form the offline training

examples of the proposed network. The training data is put into the network to classify the decision rule sets into classes according to their similarities. Then, the trained network is used to satisfy the desired performance measures determined by the FMS scheduler at specific intervals.

3 The FMS Model

The performance of the proposed approach is evaluated on an FMS. The FMS considered in this paper consists of seven jobs, six machines, one loading/unloading station and a staging area. Three AGVs are used to transport the parts within the system, each having a travel velocity of 100 feet per minute. When a vehicle completes its task and there are no other requests for transport, the vehicle is sent to the staging area to await the next request. Each job requires five operations and must visit a certain number of machines and in different sequences. The routes of each job are shown in Table 1.

Table 1. Operation sequences of each job

Job type	Operation				
	1	2	3	4	5
1	M4	M1	M4	M1	M6
2	M3	M1	M2	M3	M6
3	M5	M4	M5	M4	M6
4	M2	M3	M2	M5	M6
5	M3	M1	M4	M2	M6
6	M2	M1	M4	M2	M6
7	M3	M4	M3	M4	M6

Three decision variables are considered and different dispatching rules are examined with respect to these decision variables. Decision variables and the associated dispatching rules used are as follows: Selection of a load by an AGV (SPT- shortest processing time, FCFS-first come first served, LCFS-last come last served, EDD-earliest due date), Selection of load by machine (SPT, FCFS, EDD, CR-critical ratio, Slack, MDD-Modified due date, WSPT-Weighted shortest processing time), and Selection of an AGV by a load (Cyclical, Random, Smallest Distance First, Largest Distance First). Table 2 shows the performance measures and system status variables considered in this study.

Table 2. Evaluation criteria

Performance Measures	System status
Mean tardiness (MT)	Average number waiting
Mean flow time (MFT)	in machine queues
Total number of tardy jobs (TNTJ)	Average utilization of machines
Total weighted flow time(TWFT)	Mean waiting time in queues

The proposed network is developed using the software, NeuroSolutions 5. After training the network, classification results are obtained for 250 scenarios which include the current and next decision rule pairs to be implemented. The decision rule sets are assigned to 5 different classes. According to this, out of 250 rule sets, 113, 15, 7, 35 and 80 of them belong to class 1, class 2, class 3, class 4 and class 5, respectively. At the beginning, the system is scheduled using the randomly determined decision rules. Then, to investigate the effectiveness of the proposed approach, the manufacturing system is controlled at five different 3000 minute intervals, after a warm up period of 5000 minutes. At each control point, current performance measure values, current system status variable values, desired performance measure values of the decision maker are fed into the network to determine the class of the decision rule set to be used for the next interval. Among the decision rule sets in the determined class, the rule set which improves the performance measures the most, is chosen as the next decision rule set to be used.

4 Experimental Results

The performance of the proposed approach is evaluated at different scheduling points, relative to four different performance measures, mean flow time, mean tardiness, total number of tardy jobs, and mean weighted flow time. In the offline scheduling approach, the system is scheduled using the randomly determined rule set 3-3-2 (LCFS rule for the first decision variable, EDD rule for the second decision variable and random selection rule for the third decision variable) in all the intervals. In the proposed dynamic scheduling approach, after the current interval ends which was scheduled by the rule set 3-3-2, the next decision rules are determined by the neural network which considers the desired objectives given by the decision maker. The rule sets determined by the proposed approach are 2-4-3, 1-5-3, 2-2-3 and 4-5-2 for the second, third, fourth and the fifth interval, respectively.

From examination of the performance measures at each rescheduling point given in figure 1, it is seen that by employing the dispatching rules determined using the proposed scheduler at each period, superior solutions are obtained over the offline scheduling approach.

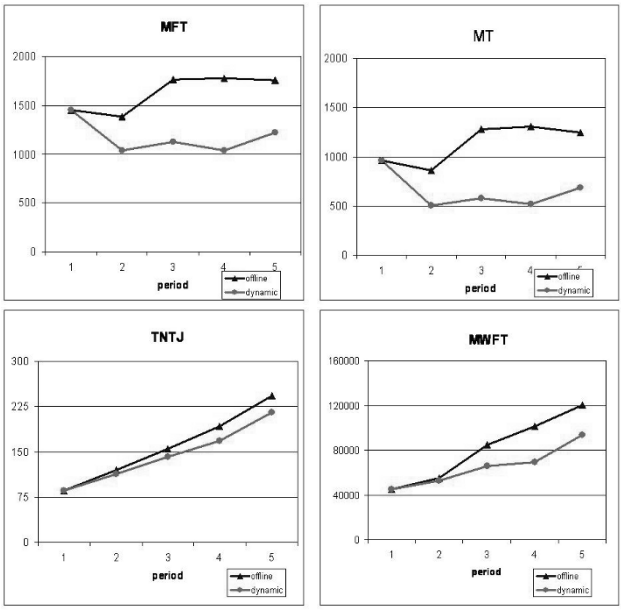


Fig. 1. Performance measures at each period (rescheduling point)

5 Conclusions and Future Research

This paper explored the combined use of competitive neural networks and simulation, as a multi-objective scheduler, to select the appropriate dispatching rules for an FMS. By the proposed method, it is possible to discover dispatching rules that will be effective for the next production interval. The results of the study showed that monitoring the system conditions at intervals and changing the rule set correspondingly, rather than using the same rule set during the whole production period, provides significant improvements in the value of system performance measures. An area for future research could be to develop a methodology to select the appropriate rule set to be used for the next interval, among the rule sets in each class. Another possible extension

of the proposed method might be, to investigate the effects of the AGVs on the selection of the next decision rules.

References

1. Akyol DE, Bayhan GM (2007) A review on evolution of production scheduling with neural networks. Computers & Industrial Engineering 53:95–122
2. Araz OU (2007) Real-Time Controlled Multi-objective Scheduling Through ANNs and Fuzzy Inference Systems: The Case of DRC Manufacturing. Lecture Notes in Computer Science 4490:973–976
3. Arzi Y, Iaroslavitz L (1999) Neural network-based adaptive production control system for a flexible manufacturing cell under a random environment. IIE Transactions 31:217–230
4. Baker KR (1984) Sequencing rules and due date assignments in a job shop. Management Science 30:1093–1103
5. Blackstone JH, Phillips DT, Hogg GL (1982) A state of the art survey of dispatching rules for manufacturing job shop operations. International Journal of Production Research 20:27–45
6. Chen W, Muraki M (1997) An action strategy generation framework for an on-line scheduling and control system in batch processes with neural networks. International Journal of Production Research 35(12):3483–3507
7. Li DC, Chen LS, Lin, YS (2003) Using Functional Virtual Population as assistance to learn scheduling knowledge in dynamic manufacturing environments. International Journal of Production Research 41(17):4011–4024.
8. Min, HS, Yih Y, Kim CO (1998) A competitive neural network approach to multi-objective FMS scheduling. International Journal of Production Research 36(7):1749–1765.
9. Min HS, Yih Y (2003) Selection of dispatching rules on multiple dispatching decision points in real-time scheduling of a semiconductor wafer fabrication system. International Journal of Production Research 41(16):3921–3941
10. Priore P, Fuente D, Pino R, Puente J (2003) Dynamic scheduling of flexible manufacturing systems using neural networks and inductive learning. Integrated Manufacturing Systems 14(2):160–168
11. Rajendran C, Holthaus O (1999) A comparative study of dispatching rules in dynamic flowshops and jobshops. European Journal of Operational Research 116:156–170
12. Vepsalainen APJ, Morton TE (1987) Priority rules for job shops with weighted tardiness costs. Management Science 33:1035–1047

Improving Classifier Performance by Using Fictitious Training Data? A Case Study

Ralf Stecking[1] and Klaus B. Schebesch[2]

[1] Faculty of Economics, University of Oldenburg, D-26111 Oldenburg, Germany. ralf.w.stecking@uni-oldenburg.de
[2] Faculty of Economics, University "Vasile Goldiş", Arad, Romania kbsbase@gmx.de

1 Introduction

Many empirical data describing features of some persons or objects with associated class labels (e.g. credit client features and the recorded defaulting behaviors in our application [5], [6]) are clearly not linearly separable. However, owing to an interplay of *relatively sparse* data (relating to high dimensional input feature spaces) and a validation procedure like *leave-one-out*, a nonlinear classification cannot, in many cases, improve this situation but in a minor way. Attributing all the remaining errors to *noise* seems rather implausible, as data recording is offline and not prone to errors of the type occurring e.g. when measuring process data with (online) sensors. Experiments with classification models on input subsets even suggest that our credit client data contain some hidden redundancy. This was not eliminated by statistical data preprocessing and leads to rather competitive validated models on input subsets and even to slightly superior results for combinations of such input subset base models [3]. These base models all reflect different views of the same data. However, class regions with highly nonlinear boundaries can also occur if important features (i.e. other explaining factors) are for some reason not available (unknown, neglected, etc.). In order to *see* this, simply project linearly separable data onto a feature subset with smaller dimension. This would account for a second type of (perceived) noise: in the case of our credit clients, besides the influence of all the client features available to a bank, the unfolding of personal events of a client may still contribute to his defaulting behavior. As such events unfold after model building, they cannot be part of the client feature data at forecasting time in any sensible way. Deciding to

which extent the errors are produced by as yet not detected nonlinearities which actually exist in a complete feature space, i.e. which are caused by an *inadequate model view* of the data and to which extent they are caused by an incomplete feature space (second type of noise) is hardly possible in practice. Besides producing more experimental evidence, adding fictitious training data may lead to making the nonlinearities more visible to the classification algorithm. In section 2 we outline a very simple placement of fictitious data using a toy model and we motivate and discuss the relation to other relevant parameters of the Support Vector Machine (SVM) classifier. In section 3 we report on using such fictitious data on our empirical credit client data. Here SVM models with different kernels are used, leading to interpretations based on certain properties of the resulting SVM, and in section 4 we conclude and address some future directions of generating fictitious training data.

2 Simple Effects of Added Fictitious Training Examples

Among nonlinear classification methods, Support Vector Machines (SVM) [4] lend themselves to interpreting the resulting model complexity by interpreting the role of the **support vectors**. Such special data points are returned by the dual SVM optimization procedure and they describe the region in input feature space where separation of classes is more difficult. Starting out with N labeled training examples $\{x_i, y_i\}_{i=1,...,N}$, with $x \in \mathcal{R}^m$ (e.g. m client features) and associated label $y_i = \{-1, 1\}$ (e.g. behavioral class of ith client), the SVM proceeds by placing a fat separating box between classes in some derived abstract space (margin maximization [4],[5],[6]). This finally leads to a separating function $S(x) = \sum_{i=1}^{N} y_i \alpha_i^* k_i(x_i, x) + b^*$, with $0 \leq \alpha_i^* \leq C$, $i = 1, ..., N$, where $C > 0$ is a user supplied control of allowable misclassification and the sign of $S(x)$ being the forecasted label of *new* points x. Kernels $k_i(.,.)$ are also user selected, a popular instance being the RBF-kernel $k_i(x_i, x) = \exp(-\sigma_i ||x_i - x||^2)$, where user supplied $\sigma_i > 0$ is case specific, emphasizing i.e. the locality of the features of the ith case. Easily separable data receive $\alpha_i^* = 0$ (they are *non support vectors*) and the more difficult (margin) cases receive $0 < \alpha_i^* \leq C$. An important difference within the set of support vectors is whether $0 < \alpha_i^* < C$ (termed *unbounded* or *essential* support vectors) or if $\alpha_i = C$ (termed *bounded* support vectors, the latter containing all cases which are *falsely separated* by $S(x)$ and hence contributing less to a useful separating function $S(x)$. Although appropriate variation of C and local σ_i leads

in principle to similar effects as can be realized by some seeding with fictitious training points, we stipulate that using such synthetic training points can be more general and has some advantages.

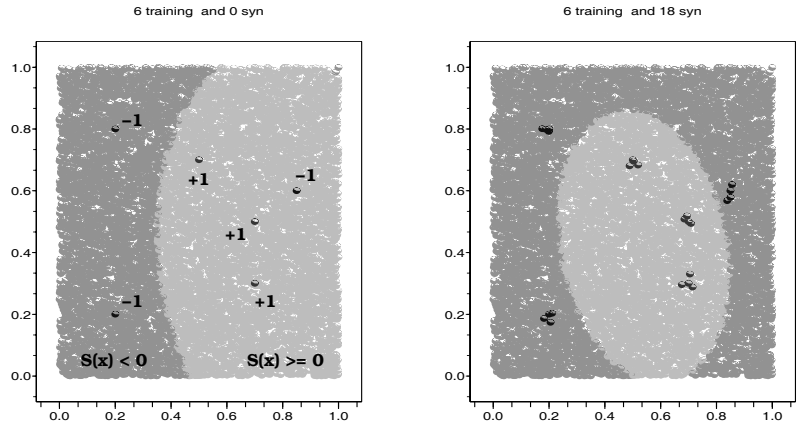


Fig. 1. A poor SVM separating function for six data points (lhs plot). Adding three fictitious points in a close vicinity of each data point leads visibly to a more adequate separation (rhs plot)

By using such additional training points, one can control both global and local effects (class-wise, case-wise and feature-wise; far beyond those studied in the present paper). Such control can be obtained by persons which are not acquainted with the internals of the SVM-method (or any other classification method) but which possess domain knowledge for producing such training points (e.g. credit officers in our application of section 3). In a toy example, we point to the simplest of all seeding by fictitious training points, namely adding a constant number of such points (indiscriminately) into the vicinity of every training point and by assigning them the same label as the respective original training point. This is done within a radius well below the minimal distance measured between any two original training points, which avoids any interference with the "nature" of the original problem and simply places "more stress" on each training example. As depicted in Fig. 1 we train a SVM on a small two dimensional example consisting of six original training points with opposite labels (signed dark points) which are not linearly separable in inputs feature space (as is the case with our empirical credit scoring data of section 3). Here we deliberately use a

C and a global $\sigma_i = \sigma$ for all $i = 1, ..., N$ within the RBF kernels such that a separating function $S(x) = \sum_{i=1}^{6} y_i \alpha_i^* \exp(-\sigma ||x_i - x||^2) + b^*$ results, which is visibly a poor solution (in the figure, the two different background shades indicate whether $S(x) \geq 0$ or $S(x) < 0$ for $x \in [0, 1] \times [0, 1]$). By adding a set of fictitious training points in the vicinity of each original training point (without changing any internals of the SVM) the solution *jumps* to a much better separating function (rhs plot). We note this effect can be in fact achieved by placing such fictitious points arbitrarily close to their original points.

3 Fictitious Points for Different Kernels on Real Data

The real data set for our credit scoring models is a sample of 658 clients for a building and loan credit with a total number of 40 input variables. It contains 323 defaulting and 335 non defaulting credit clients [5]. Fictitious data points are calculated by generating normally distributed random variables with zero mean and a standard deviation of $s = 0.059$ (which is half of the minimum distance between the credit client data points). These random numbers are added to the real data values. To each of the 658 credit clients *five* of these fictitious data points are added, resulting in a final data set, consisting of 3948 (658 original plus 3290 fictitious) data points. SVM with five different kernel functions are then used for classifying good and bad credit clients. Detailed information about kernels, hyperparameters and tuning can be found in [6].

In table 1 for each kernel the number of support vectors (SV), the number of bounded support vectors (BSV), the vector norm and the training error is shown for models trained on the full fictitious data set and on the the real credit scoring data set. In general, using fictitious data seems to affect highly non linear kernel functions like polynomial (especially 3$^{\text{rd}}$ degree) and RBF kernels stronger than linear and sigmoid kernels. We detect only small differences in the training error as well as in the vector norm for linear and sigmoid kernel when changing from real data models to real plus fictitious data models. The polynomial kernels and the RBF kernel, on the other hand, show huge differences in training error and vector norm when comparing both data sets. Especially the small amount of bounded support vectors for these models indicates an improvement of the generalization error [2].

Table 1. Evaluation and comparison of five SVM with different kernel functions. Each model is trained and evaluated on a full fictitious data set (*Full Fict.*), containing 658 real data points together with 3290 fictitious data points in close neighborhood. *Real data* model is trained and evaluated without using any additional fictitious data points, *FF subset* is trained on the full fictitious data set and evaluated on the real data set

SVM-Kernel	No. of Cases	No. of SV	No. of BSV	SV+ BSV	Vector Norm	Training Error
Linear						
Full Fict.	3948	47	2012	2059	3.8535	22.92 %
FF subset	658	5	336	341		22.80 %
Real data	658	41	316	357	4.0962	22.64 %
Sigmoid						
Full Fict.	3948	21	2568	2589	11.8922	25.97 %
FF subset	658	4	428	432		25.84 %
Real data	658	17	544	561	10.5478	25.84 %
Polyn. 2^{nd} deg.						
Full Fict.	3948	159	1877	2036	1.3913	12.41 %
FF subset	658	20	321	341		11.85 %
Real data	658	63	392	455	0.5868	19.60 %
Polyn. 3^{rd} deg.						
Full Fict.	3948	433	845	1278	0.8494	1.85 %
FF subset	658	47	135	182		1.82 %
Real data	658	216	211	427	0.3909	8.81 %
RBF						
Full Fict.	3948	412	1104	1516	63.5659	3.29 %
FF subset	658	51	173	224		3.34 %
Real data	658	179	252	431	26.6666	10.94 %

4 Conclusions and Outlook

In this contribution we started investigating the effects of using fictitious training examples to a credit scoring problem, which we intensively studied by means of SVM modeling in previous work. After stating potential advantages of controlling the separating function by this type of intervention as opposed to manipulating SVM internals, we investigate the effect of the simplest placement of fictitious training points, namely seeding the vicinity of each training point with randomly drawn points having the same label as the original data point.

Applying this to the intensively trained SVM models with different kernels we observe that the linear models do not change much (as would be expected by theory) but some non-linear models show some interesting change in model capacity (relatively more parsimony), which still needs further investigation.

Adding fictitious training data also points to the problem of whether *newly generated* training points do actually express feasible domain data. In the context of credit scoring and in many other classification problems it is quite obvious that *not every* combination of input features can be a feasible case description (e.g. a client) for any of the classes. A profound debate in classification is connected to the importance of this *generative modeling*, especially relating to the question of whether a good model should be able to generate a large number of feasible class members from a small set of initial examples [1]. Future work will use more refined seeding, generating fictitious credit clients, e.g. with feasible ordinal feature instances, etc., to be used in complement with the original training data.

References

1. Duin, R.P.W. and Pekalska, E. (2005): Open issues in pattern recognition, to be found at:
`www-ict.ewi.tudelf.nl/~duin/papers/cores_05_open_issues.pdf`
2. Schebesch, K.B. and Stecking, R. (2007): Selecting SVM Kernels and Input Variable Subsets in Credit Scoring Models. In: Decker, R., Lenz, H.-J. (Eds.): *Advances in Data Analysis*. Springer, Berlin, 179–186.
3. Schebesch, K.B. and Stecking, R. (2007): Using Multiple SVM Models for Unbalanced Credit Scoring Data Sets. Proceedings of the 31th International GfKl Conference, Freiburg.
4. Schölkopf, B. and Smola, A. (2002): *Learning with Kernels*. The MIT Press, Cambridge.
5. Stecking, R. and Schebesch, K.B. (2003): Support Vector Machines for Credit Scoring: Comparing to and Combining with some Traditional Classification Methods. In: Schader, M., Gaul, W., Vichi, M. (Eds.): *Between Data Science and Applied Data Analysis*. Springer, Berlin, 604–612.
6. Stecking, R. and Schebesch, K.B. (2006): Comparing and Selecting SVM-Kernels for Credit Scoring. In: Spiliopoulou, M., Kruse, R., Borgelt, C., Nürnberger, A., Gaul, W. (Eds.): *From Data and Information Analysis to Knowledge Engineering*. Springer, Berlin, 542–549.

Part V

Continuous Optimization

Artificial DMUs and Contingent Weight Restrictions for the Analysis of Brazilian Retail Banks Efficiency

Madiagne Diallo[1], Marcus Vinicius Pereira de Souza[2], Luis Eduardo Guedes[3], and Reinaldo Castro Souza[2]

[1] Departamento de Engenharia Industrial, Pontifícia Universidade Católica Rio de Janeiro, Brazil. diallo@ind.puc-rio.br
[2] Departamento de Engenharia Elétrica, Pontifícia Universidade Católica Rio de Janeiro. mvinic@hotmail.com, reinaldo@ele.puc-rio.br
[3] Departamento de Engenharia de Produção, Universidade Federal do Rio de Janeiro, Brazil. guedes1970@uol.com.br

Summary. In this paper, we deal with the BCC-I model in Data Envelopment Analysis comparing results of Artificial DMUs and the adjusted Contingent Weight Restrictions approach in the analysis of Brazilian Retail Banks efficiencies. The number of employees, fixed assets, leverage and delinquency rate were considered as inputs, and the financial intermediation results and the equities profitability as outputs. While the Contingent Weight Restrictions method makes only directed weight restrictions, Artificial DMUs method simulates a set of weight restrictions and depends on the use of specialists' opinions. The compared methods are relevant to increase the precision of the market analysis. The results of both methods are compared in order to distinguish their advantages and inconveniences. A set of efficient banks is obtained together with a market analysis.

Key words: Data Envelopment Analysis, Finance and Banking, Multi-Criteria Decision Aids, Artificial DMUs, Contingent Weight Restrictions.

1 Introduction

The Brazilian economy is achieving a stable and robust condition. In addition, the main Central Bank Interest Rate decreased 56.6% from October 2002 to June 2007, from 26.5% to 11.5% per year, encouraging innovative and attractive popular credit offers. This drastically increased the competition in the banking market. Thus, it is interesting for each retail bank to identify with precision variables that could

impact the market share. The analysis using financial indices helps the analyst to evaluate the financial health of an organization and enables the perception of the weak and strong points related to its structure, liquidity, profitability and activity. The aim of this work was to analyze the efficiency of the 21 largest Brazilian Retail Banks using the BCC-I model in Data Envelopment Analysis (DEA), comparing results provided by the artificial DMUs method and the adjusted Contingent Weight Restrictions approach. While the Contingent Weight Restrictions method makes only directed restrictions on weights, Artificial DMUs method simulates a set of weight restrictions and depends on the use of the specialists' opinions. For the purpose of the analysis, 6 variables composed of 4 indices as input variables and 2 indices as output variables are defined. The selection of variables includes financial indices with the characteristic of *the lower the better* for representing the input variables, and financial indices with the characteristic of *the higher the better*, for representing the output variables. The data used in this work are real and available at Brazilian Central Bank http://www.bcb.gov.br/fis/top50/port/default.asp?parmidioma=P&id=top50.

2 Description of Variables and Methodology

The application of DEA on the analysis of accountancy report tackles with the negative values that accountancy results assume. The translation invariance approach discussed in [4] allows overcoming such difficulty. Thus, depending on the case studied, neither the negative values are converted into positive without impact on the efficiency analysis, or the inefficiency and efficiency classification is adapted. In our case, for some banks one or both output variable may be concerned. **Input variables**: **(1) Number of employees**: represents the total number of persons with employment contract directly made with the respective bank. People working through outsourcing contracts are not taken into account.**(2) Leverage**: indicates the relation between the resources of third and the bank's proper capital. It measures the aggressiveness of the institution. Higher its indice is, more is risk envolved in the bank's operations. **(3) Delinquency rate**: indicates the relation between the amount of credits in liquidation and the total value of credit provided by the bank. It measures the amount of loans difficult to be paid back. The lower, the better. **(4) Fixed asset**: indicates the proportion of capital invested in permanent assets. The lower, the better. **Output variables**: **(1) Financial Intermediation result**: corresponds to the difference between revenues and expenses issued

from capital markets, movable headings and values, change, compulsory applications and others.**(2) Equity's profitability**: measures the final return of stockholders regarding the own capital of the bank. The higher, the better.

2.1 Artificial DMUs Method

The flexibility, considering the existence of weights in the classic DEA method, is important for the identification of inefficient DMUs, i.e, The one presenting low performance even with weights favorably defined. However, in DEA, weights allocation is a complex task and a bad choice of weights introduced in the Linear Program restrictions may lead the problem to become unfeasible. The authors in [6] established that each weight in DEA, strictly positive, is equivalent to a none observed DMU (*artificial* DMU), introduced among the others during the analysis. The observation was generalized in [1] for the case of multiple inputs and/or outputs, applied to DMUs that operate with constant return scale or to those operating with variable return scale. In this context, the inclusion of an artificial DMU in the original set of DMUs works as an alternative method for the simulation of a set of weight restrictions, where the efficiency indices relative to this new set are computed using the classic BBC-I model that does not require weight restrictions. The coordinates selected for the artificial DMUs are fundamental for the solution effectiveness. The artificial DMUs can be defined using equations (II.1) or (II.2), without impact on the results.

$$y_{rjt} = \frac{y_{rj}}{h_j^*} \text{ and } x_{ijt} = x_{iij}, \forall jt = j \quad \textbf{(II.1)}$$
$$y_{rjt} = y_{rj} \text{ and } x_{ijt} = x_{ij} \times h_j^*, \forall jt = j \quad \textbf{(II.2)}$$

As for the classic BBC-I model, efficiency depends on the orientation of the model. Thus, the definition of the artificial DMU using contraction of inputs as expressed in equations **(II.3)**, does not produce the same results if the expansion of outputs expressed in **(II.4)** is used.

$$y_{rjv} = y_{rj} \text{ and } x_{ijv} = x_{ij} \times v_i^*, \forall jv = j \quad \textbf{(II.3)}$$
$$y_{rjv} = \frac{y_{rj}}{v_j^*} \text{ and } x_{ijv} = x_{ij}, \forall jv = j \quad \textbf{(II.4)}$$

In the particular case of this work, the average equity's profitability has been defined as an efficiency cutting criteria. Thus, it has been established that no bank with an equity's profitability below the average would be more efficient than another bank with an equity's profitability above it.

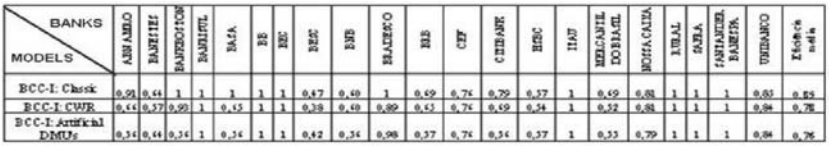
2.2 Adjusted Contingent Weight Restrictions Approach

As illustrated in [1], there is a large variety of analyses relative to weight restrictions, making it one of the most promising in the DEA theory. In the basic multiplier models (CCR and BCC) [2], it was observed that the weights u and v are variables restricted to be greater than or iqual to an infinitesimal positive value ϵ so that no input or output value could be totally ignored in the computation of the respective DMUs efficiencies. Allowing some flexibility in the selection of weights is frequently presented as an advantage in DEA applications [2]. A priori, weight specification is in fact not required and each DMU is evaluated to its best performance. However, in some situations, this complete flexibility can give margin to undesirable consequences. After all, one may evaluate a DMU as efficient in situations difficult to justify. Imposing weight restrictions allows to incorporate some information based on the specialists's opinions, preferences on management style, or other judgments. Therefore, the DEA model becomes more plausible and a more coherent analysis about the performance of the DMUs is obtained. As another alternative, the goal could be to better reflect the objectives or values of the organizations. In most cases, this process presents an important challenge for the analyst who would need to explain the reason for which his company becomes inefficient when weight restrictions are introduced in the model. However, it is important to stress that there are controversies about these weight restrictions aspects. In [3], it is advocated that the results obtained from the models with weight restrictions cannot be interpreted in the same way as if they were obtained with the original models. This occurs when weight restrictions are imposed, the interpretation of the Production Possibility Set becomes invalid. The characteristic of the radial model is also lost. In [5] it is argued that the weight restrictions should be imposed taking into consideration the input and output levels of each DMU. Hence, it is insured that only the inputs or outputs that in fact contribute significantly for the performance of a DMU are included in the analysis. In [5], it is proposed to use restrictions not on v_i, but rather on $v_i X_i$, where v_i is the weight of input i and $v_i X_i$ is the product of input i with its weight. The mathematic model is: $v_i X_i \leq \mathbf{k} v_i X_i, \quad (i \neq j)$ The approach firstly models restrictions taking into account only the DMUs of efficiency 1 using the classic DEA model. Then, the DMU that appears on the left side of the inequation is the one which was more frequently peer-grouped. In this case, 6 blocks with 9 restrictions each ("Assurance Regions Type II") like in [7] are added to the model. As

for the value of variable **k**, it is obtained from a simple linear regression between two inputs.

3 Analysis of Results and Comparisons

Fig. 1 summarizes the results obtained with the Classical BCC-I, adjusted CWR and artificial DMUs methods. Analyzing the results, one



BANKS / MODELS	ABN AMRO	BANRISUL	BANRISUL	BASA	BB	BEC	BESC	BNB	BRADESCO	BSB	CEF	CITIBANK	HSBC	ITAU	MERCANTIL DO BRASIL	MONEO CAIXA	RURAL	SAFRA	SANTANDER BANESPA	UNIBANCO	Eficiência média
BCC-I: Classic	0,98	0,44	1	1	1	1	1	0,47	0,40	1	0,69	0,76	0,79	0,57	1	0,69	0,81	1	1	1	0,83	0,85
BCC-I: CWR	0,44	0,37	0,99	1	0,45	1	1	0,38	0,40	0,89	0,63	0,76	0,69	0,54	1	0,52	0,81	1	1	1	0,84	0,78
BCC-I: Artificial DMUs	0,34	0,44	0,34	1	0,34	1	1	0,42	0,34	0,98	0,37	0,76	0,34	0,37	1	0,33	0,79	1	1	1	0,84	0,76

Fig. 1. Results of the Classical BCC-I, adjusted CWR and artificial DMUs methods

can see that the classical model (no weight restrictions), considered 9 efficient DMUs, while the adjusted CWR and the artificial DMUs methods considered only 7 efficient DMUs, being thus more discriminatory. Surprisingly, both adjusted CWR and artificial DMUs methods found an iqual set of efficient DMUs. With regard to the comparison of the adjusted CWR and artificial DMUs methods discrimination powers, it was concluded that the artificial DMUs method is more discriminatory showing a lower efficiency average. All the DMUs have recorded at least

Banks	BANRISUL	BB	BEC	ITAU	RURAL	SAFRA	SANTANDER
Number of employees	HC	NC	NC	HC	HC	HC	NC
leverage	NC	NC	NC	NC	NC	NC	NC
Delinquency rate	NC	NC	NC	LC	NC	NC	HC
Fixed Assets	NC	HC	HC	NC	NC	NC	NC
Financial intermediation	NC	HC	NC	HC	HC	HC	LC
Equity's Profitability	HC	NC	LC	LC	HC	MC	HC

Fig. 2. Percentage of units by efficiency level:(HC - H=High, M=Medium, L=Low, N=None and C means Contribution. Ex: HC=High Contribution)

one variable with weight 0 (NC), *i.e*, variable ignored in the DMU's efficiency computation. Probably because, if it was considered, the Bank (DMU) would become efficient. Or it may be that the solution found by the model is the one that does not consider null weights for any variables, and it may exist, in case that the DMU is considered in fact efficient. The market analysis is based on the virtual participation of variables in the computation of efficiencies. This analysis is only based

on the artificial DMU approach, since in the adjusted CWR, the weight restrictions is direct and does not allow such analysis.

Common market strategies of efficient banks: focus on consumption credits (cf. high equity's profitability in Fig. (2); association or acquisition with credit specialty institutions, operations with headings (cf. high Financial intermediation in Fig. (2).

4 Conclusion

The application of the artificial DMUs method in substitution to a set of weight restrictions proved viability in the case here analyzed, since aggregated opinions of specialists have come to the same conclusions. The adjusted CWR approach has also shown efficiency since it increased the discrimination power with respect to the classical model, and has come to the result of artificial DMU method. However, the adjusted CWR does not provide information for market analysis, since it does not generate the virtual participations of the variables in the computation of efficiencies. The adjusted CWR approach seemed to be more appropriate for problems that require less interference from specialists such as pricing and regulations. For problems requiring a deep analysis of the variables involved, artificial DMUs is more suitable.

References

1. Allen, R., et al. (1997), *Weight restrictions and value judgements in data envelopment analysis: evolution, development and future directions.* Annals of Operations Research, 73:13–34.
2. Charnes, A., et al. (1994), *Data envelopment analysis: theory, methodology, and application.* Boston, MA: Kluwer Academic.
3. Dyson, R. G., et al. (2001), *Pitfalls and protocols in DEA.* European Journal of Operational Research, 132:245–259.
4. Pastor, J. T. (1997) *Translation invariance in data envelopment analysis: a generalization.* Annals of Operation Research, 73:91–115.
5. Pedraja-Chaparro, F., Salinas-Jimenez, J., Smith, P. (1997), *On the role of Weight Restrictions in Data Envelopment Analyis.* Journal of Productivity Analysis, 8:215–230.
6. Roll, Y., Golany, B. (1991), *Controlling factor weights in DEA.* IIE Transaction, 23(1):2–9.
7. Thanassoulis, A., Allen, R. (1998), *Simulating Weight Restrictions in Data Envelopment Analysis by Means of Unobserved DMUs* Management Science, 44:586–594, 1998.

Performance of Some Approximate Subgradient Methods over Nonlinearly Constrained Networks*

Eugenio Mijangos

University of the Basque Country, Department of Applied Mathematics and Statistics and Operations Research, P.O. Box 644, 48080 Bilbao, Spain
eugenio.mijangos@ehu.es

Summary. Nonlinearly constrained network flow problems can be solved by using approximate subgradient methods. This work studies the influence of some parameters over the performance of some approximate subgradient methods and compares their efficiency with that of other well-known codes. Numerical results appear promising.

Key words: Nonlinear Programming, Subgradient Methods, Networks

1 Introduction

Consider the nonlinearly constrained network flow problem **(NCNFP)**

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) && (1)\\ \text{subject to} \quad & x \in \mathcal{F} && (2)\\ & c(x) \leq 0, && (3) \end{aligned}$$

where we assume throughout this paper that:

- $\mathcal{F} = \{x \in \mathbb{R}^n \mid Ax = b,\ 0 \leq x \leq \overline{x}\}$, A is a node-arc incidence $m \times n$-matrix, b is the production/demand m-vector, x are the flows on the arcs, and $\overline{x}$ are the capacity bounds.
- The side constraints (3) are defined by $c : \mathbb{R}^n \to \mathbb{R}^r$, where the components are nonlinear and twice continuously differentiable on $\mathcal{F}$.
- $f : \mathbb{R}^n \to \mathbb{R}$ is nonlinear and twice continuously differentiable on $\mathcal{F}$.

* The research was partially supported by grant MCYT DPI 2005-09117-C02-01

In this work we focus on the primal problem **NCNFP** and its dual problem

$$\text{maximize} \quad q(\mu) = \min_{x \in \mathcal{F}} \; l(x, \mu) = \min_{x \in \mathcal{F}} \; \{f(x) + \mu^t c(x)\} \tag{4}$$

$$\text{subject to:} \quad \mu \in \mathcal{M}, \tag{5}$$

where $\mathcal{M} = \{\mu \mid \mu \geq 0, \; q(\mu) > -\infty\}$. We assume throughout this paper that the constraint set $\mathcal{M}$ is closed and convex, q is continuous on $\mathcal{M}$, and for every $\mu \in \mathcal{M}$ some vector $x(\mu)$ that minimizes $l(x, \mu)$ over $x \in \mathcal{F}$ can be calculated, yielding a subgradient $c[x(\mu)]$ of q at μ.

When, as happens in this work, for a given $\mu \in \mathcal{M}$, the dual function value $q(\mu)$ is calculated by minimizing approximately $l(x, \mu)$ over $x \in \mathcal{F}$, the computation of the subgradient and $q(\mu)$ involves an error. Given a scalar $\varepsilon \geq 0$ and a vector $\overline{\mu}$ with $q(\overline{\mu}) > -\infty$, we say that c is an *ε-subgradient at $\overline{\mu}$* if $q(\mu) \leq q(\overline{\mu}) + \varepsilon + c^t(\mu - \overline{\mu})$, for all $\mu \in \mathbb{R}^r$. The set of all ε-subgradients at $\overline{\mu}$ is called the *ε-subdifferential at $\overline{\mu}$* and is denoted by $\partial_\varepsilon q(\overline{\mu})$.

An *approximate subgradient method* is defined by

$$\mu^{k+1} = [\mu^k + s_k c^k]^+, \tag{6}$$

where c^k is an ε_k-subgradient at μ^k, $[\cdot]^+$ denotes the projection on the closed convex set $\mathcal{M}$, and s_k is a positive stepsize.

In our context, we minimize approximately $l(x, \mu^k)$ over $x \in \mathcal{F}$ by efficient techniques specialized for networks [11], thereby obtaining a vector $x^k \in \mathcal{F}$ with $l(x^k, \mu^k) \leq \inf_{x \in \mathcal{F}} l(x, \mu^k) + \varepsilon_k$.

This work studies the influence of some parameters over the performance of several approximate subgradient methods in the solution of **NCNFP**. Furthermore, the efficiency of the implementation of these methods is compared with that of filterSQP [5] and KNITRO [2].

The remainder of this paper is structured as follows. Section 2 presents the ways of computing the stepsize in the approximate subgradient methods; Section 3 describes the solution to the nonlinearly constrained network flow problem; and Section 4 puts forward experimental results.

2 Calculation of the Stepsizes

2.1 Diminishing Stepsize Rule (DSR)

The convergence of the exact subgradient method using a diminishing stepsize was shown by Correa and Lemaréchal [3]. Since c^k is an ap-

proximate subgradient, the convergence is analyzed by Proposition 2.3 in [6]. An example of such a stepsize is $s^k = s/\widehat{k}$, for $\widehat{k} = \lfloor k/m \rfloor + 1$ with $s > 0$. We use by default $m = 5$ and $s = 100$.

2.2 Variant of the Constant Step (VCS)

As is well known the classical scaling of Shor (see [10]) $s_k = s/\|c^k\|$ gives rise to a s-constant-step algorithm.

In our case c^k is an approximate subgradient, hence it can exist a k such that $c^k \in \partial_{\varepsilon_k} q(\mu^k)$ with $\|c^k\| = 0$, but ε_k not being sufficiently small. In order to overcome this trouble we have considered the variant $s_k = s/(\delta + \|c^k\|)$, where s and δ are positive constants. The convergence of this variant is analyzed by Proposition 1 in [7]. In this work by default $\delta = 10^{-12}$, with $s = 100$.

2.3 Dynamic Stepsize with Adjustment Procedure (DSAP)

An interesting alternative for the ordinary subgradient method is the *dynamic stepsize rule* (see [9]), where $s_k = \gamma_k[q^* - q(\mu^k)]/\|c^k\|^2$. In our case, $c^k \in \partial_{\varepsilon_k} q(\mu^k)$, $q(\mu^k)$ is approximated by $q_{\varepsilon_k}(\mu^k) = l(x^k, \mu^k)$, and q^* is replaced with an estimate q^k_{lev}.

In this procedure (see [8] and [6]) q^k_{lev} is the best function value achieved up to the kth iteration, in our case $\max_{0 \le j \le k} q_{\varepsilon_j}(\mu^j)$, plus a positive amount δ_k, which is adjusted according to algorithm's progress, i.e.

$$\delta_{k+1} = \begin{cases} \rho\delta_k, & if\, q_{\varepsilon_{k+1}}(\mu^{k+1}) \ge q^k_{lev}, \\ \max\{\beta\delta_k, \delta\}, & if\, q_{\varepsilon_{k+1}}(\mu^{k+1}) < q^k_{lev}, \end{cases} \tag{7}$$

where δ_0, δ, β, and ρ are fixed positive constants with $\beta < 1$ and $\rho \ge 1$. The convergence of the ε-subgradient method for this stepsize type is analyzed by Proposition 2.5 in [6]. In this work by default $\delta = 10^{-7}|l(\mu^1, x^1)|$, $\delta_0 = 0.5\|c(x^1)\|$, $\beta = 1/\rho$, and $\rho = 1.2$.

3 Solution to NCNFP

In order to solve **NCNFP** an algorithm is put forward below, which uses the approximate subgradient methods given by (6) and the stepsizes described in Section 2. The value of the dual function $q(\mu^k)$ is estimated by minimizing approximately $l(x, \mu^k)$ over $x \in \mathcal{F}$ so that the optimality tolerance becomes more rigorous as k increases (see [1]) and $q_{\varepsilon_k}(\mu^k) = l(x^k, \mu^k)$, where x^k minimizes approximately the nonlinear network subproblem $\mathbf{NNS_k}$ given by $\min_{x \in \mathcal{F}} l(x, \mu^k)$.

Algorithm 1

Step 0 *Initialize.* Set $k = 1$ and $\mu^1 = 0$.

Step 1 *Compute* the dual function estimate, $q_{\varepsilon_k}(\mu^k)$, by solving $\mathbf{NNS_k}$, so that if $x^k \in \mathcal{F}$ is an approximate solution, $q_{\varepsilon_k}(\mu^k) = l(x^k, \mu^k)$, and $c^k = c(x^k)$ is an ε_k-subgradient of q in μ^k.

Step 2 *Check the stopping rules* for μ^k. Without a duality gap, (x^k, μ^k) is a primal-dual solution.

Step 3 *Update* the estimate μ^k by means of the iteration (6), where s_k is computed using some stepsize rule from among DSR, VCS, and DSAP. Go to Step 1.

The implementation in Fortran-77 of the previous algorithm, termed PFNRN05, was designed to solve large-scale nonlinear network flow problems with nonlinear side constraints. More details in [6].

4 Numerical Tests

In order to evaluate the performance of PFNRN05 with the different stepsizes, some numerical tests have been carried out with **NCNFP** problems. These test problems have until 4008 variables, 1200 nodes, and 1253 side constraints. The objective functions are nonlinear and convex, and are either Namur functions (**n1**) or polynomial functions (**e2**). The side constraints are defined by convex quadratic functions. More details in [7].

In Tables 1–3 heading "$\widehat{c}$" means the opposite value of the decimal logarithm of the maximum violation of the side constraints at the optimizer $\overline{x}$ (i.e. $\widehat{c} = -\log_{10} \|c(\overline{x})\|_\infty$), "t" is the run time in seconds, "iit" indicates the number of inner iterations, and "κ" represents the decimal logarithm of an estimate of the conditioning at the optimizer.

After carrying out a series of tests over different problems, the study of the parameters has given rise to some conclusions. For VCS, if δ is increased, the iteration number increases and the solution quality is reduced (see Table 1). For $\delta < 10^{-12}$ the changes are insignificant. For DSR, if s is increased, the solution quality is slightly better, but the numerical condition worsens clearly (see Table 2). For DSAP, as ρ is moved away from 1 the solution quality gets worse, except for c24e2 with $\rho = 1.2$ (see Table 3).

In Table 4 the efficiency of our code when using VCS, DSR, and DSAP is compared with that of KNITRO [2] and filterSQP [5] by means of the run-times in CPU-seconds. These last two solvers are available on the NEOS server [4]. PFNRN is executed on a Sun Sparc 10/41

Table 1. Study of δ on VCS ($s = 100$)

	$\delta = 0$		$\delta = 10^{-16}$		$\delta = 10^{-4}$		$\delta = 1$		$\delta = 10^2$	
prob	$\widehat{c}$	t	$\widehat{c}$	t	$\widehat{c}$	t	$\widehat{c}$	t	$\widehat{c}$	t
c13e2	5	1.2	5	1.2	5	2.0	3	5.3	2	10.4
c15e2	6	4.5	6	4.5	5	6.2	4	24.4	2	36.9
c13n1	6	107.5	6	105.2	6	117.4	5	86.6	3	57.4
c15n1	6	77.9	6	75.9	6	78.4	4	331.6	3	596.3
c23e2	5	6.3	5	6.2	5	6.2	3	6.8	2	10.1

Table 2. Study of s on DSR

	$s = 1$			$s = 10$			$s = 10^2$			$s = 10^3$			$s = 10^4$		
prob	κ	$\widehat{c}$	t	κ	$\widehat{c}$	t	κ	$\widehat{c}$	t	κ	$\widehat{c}$	t	κ	$\widehat{c}$	t
c13e2	4	3	0.6	5	5	0.8	6	4	0.7	7	6	1.6	8	8	1.1
c15e2	5	1	11.1	5	2	2.2	7	4	2.7	8	5	5.4	8	8	33.6
c13n1	5	3	33.5	6	4	48.9	7	5	84.8	8	7	56.0	9	–	–
c15n1	7	3	99.2	8	4	92.3	8	4	431.0	8	–	–	10	–	–
c23e2	4	1	6.1	6	2	5.5	6	3	5.0	7	5	7.2	8	7	7.9

Table 3. Study of ρ on DSAP

	$\rho = 1$		$\rho = 1.1$		$\rho = 1.2$		$\rho = 2$		$\rho = 3$		$\rho = 10$	
prob	$\widehat{c}$	iit	$\widehat{c}$	iit	$\widehat{c}$	iit	$\widehat{c}$	iit	$\widehat{c}$	iit	$\widehat{c}$	iit
c13e2	9	635	8	628	8	623	8	594	8	571	8	581
c15e2	7	1200	7	1230	6	1131	1	876	-1	562	-2	494
c13n1	7	2313	7	2357	7	2658	6	2270	6	2462	6	2620
c15n1	9	5444	9	4648	8	4413	7	5517	6	4163	1	3084
c23e2	7	1688	7	1505	7	1557	7	1656	7	1659	7	1956
c24e2	6	2957	6	2703	10	3238	-1	1673	-1	1588	-2	1564

work station under UNIX with a similar speed to that of the NEOS machines. The value of the solution quality parameter $\widehat{c}$ appears in parentheses together with the time. As can be observed the solution quality of DSR is significantly worse than that obtained with VCS, and that of this is slightly worse than that of DSAP.

We observe that while KNITRO has been more robust, filterSQP has been more efficient except for the problem c12n1. In most problems VCS and DSAP have been the most efficient and, if we also take into account the solution quality (in parentheses), the best performance has been obtained by DSAP. These results encourage to carry out further experimentation with other stepsize types and with real problems.

Table 4. Comparison of the efficiency

prob	KNITRO	filterSQP	VCS	DSR	DSAP
c13e2	65.9	5.2	1.1 (5)	0.7 (4)	0.7 (8)
c15e2	819.0	11.5	4.4 (5)	2.7 (4)	1.6 (6)
c17e2	311.1	15.8	6.1 (5)	6.4 (4)	2.4 (5)
c12n1	10.0	362.0	59.5 (8)	81.0 (6)	57.4 (10)
c13n1	2883.6	550.3	115.5 (6)	84.8 (5)	68.8 (7)
c15n1	728.1	–	80.1 (6)	431.0 (4)	97.8 (8)
c17n1	971.8	–	78.8 (6)	358.8 (4)	198.8 (10)
c22e2	418.0	60.8	4.4 (6)	2.5 (4)	2.1 (6)
c23e2	461.9	98.5	6.5 (5)	5.0 (3)	6.9 (7)
c24e2	13482.5	–	15.4 (5)	40.0 (3)	5.1 (10)
c32e2	38.6	16.9	1.1 (9)	1.5 (9)	1.1 (8)
c33e2	2642.4	48.0	1.5 (6)	2.0 (7)	1.6 (7)
c35e2	10642.3	57.5	2.3 (6)	2.9 (7)	2.3 (9)

References

1. Bertsekas D.P. (1999) Nonlinear Programming: Second Edition. Athena Scientific, Belmont, Massachusetts
2. Byrd R.H., Nocedal J., Waltz R.A. (2006) KNITRO: An Integrated Package for Nonlinear Optimization. In: G. di Pillo and M. Roma (eds) Large-Scale Nonlinear Optimization, Springer-Verlag
3. Correa R., Lemaréchal C. (1993) Convergence of some algorithms for convex minimization. Mathematical Programming 62:261–275.
4. Czyzyk J., Mesnier M.P., Moré J. (1998) The NEOS server. IEEE Computational Science and Engineering 5(3):68–75.
5. Fletcher R., Leyffer S. (1998) User manual for filterSQP. University of Dundee Numerical Analysis Report NA\181
6. Mijangos E. (2006) Approximate subgradient methods for nonlinearly constrained network flow problems. Journal of Optimization Theory and Applications 128(1):167–190
7. Mijangos E. (2006) A variant of the constant step rule for approximate subgradient methods over nonlinear networks. In: Proc. ICCSA 2006, Lecture Notes in Computer Science 3982:767–776
8. Nedić A., Bertsekas D.P. (2001) Incremental subgradient methods for nondifferentiable optimization. SIAM Journal on Optimization 12(1):109–138
9. Poljak B.T. (1969) Minimization of unsmooth functionals. Z. Vyschisl. Mat. i Mat. Fiz. 9:509–521
10. Shor N.Z. (1985) Minimization methods for nondifferentiable functions. Springer-Verlag, Berlin Heidelberg New York
11. Toint Ph.L., Tuyttens D. (1990) On large scale nonlinear network optimization. Mathematical Programming 48:125–159

Part VI

Discrete and Combinatorial Optimization

Shortest-Path Algorithms and Dynamic Cost Changes

Sven Baselau, Felix Hahne, and Klaus Ambrosi

Institut für Betriebswirtschaft und Wirtschaftinformatik, Universität Hildesheim

Summary. Shortest-path algorithms are used to find an optimal way through a network. These networks often underlie dynamic changes, e.g. in a road network we find congestions or road works. These dynamic changes can cause a previously calculated route to be not up-to-date anymore. A shortest-path algorithm should react on these changes and present a new route without much overhead in time and space. The simplest way would be to calculate the whole route again. Dynamic shortest path algorithms with different features have been developed avoiding a full re-calculation. This paper describes the advantages of dynamic algorithms and provides an overview.

1 Introduction

Shortest-path algorithms are a widely studied field where a lot of different characteristics have been developed. Part of the research in the last years brought up dynamic shortest-path algorithm (sp-algorithm) which can handle edge cost changes without re-calculating the whole route(s) from scratch. An application for dynamic algorithms is the road network where congestions can lead to a cost increase and if a congestion disappears the costs need to be reset. A cost change in a network may have impacts on a previously calculated route, it might not be optimal anymore. Sp-algorithms should take this into account and present a new optimal route as efficiently as possible.

2 Shortest-Path Algorithms in Dynamic Domains

This section deals with a detailed view on sp-algorithms used in dynamic domains. In the following we expect only positive link costs and all algorithms are able to handle cost increases and cost decreases.

Sp-algorithms used in dynamic domains can be split up into static and dynamic algorithms. Static algorithms are characterized by re-calculating a route after a cost change from scratch. The most popular algorithms are the Dijkstra, Bellman-Ford-Moore and A* algorithm (see [1]). The main drawback of a full re-calculation is the overhead of the calculation if the affected part of the network is relatively small. All nodes which need to be updated after a cost change are called affected nodes. Static algorithms will re-calculate all node distances even if a node is not affected.

Dynamic sp-algorithms try to avoid these unnecessary calculations. Their intention is to use additional information for the determination of the affected and non-affected parts of the network. They try to update only the affected parts and to adopt the distance for each non-affected node as it stands. This approach leads to a faster calculation of a new optimal route in most cases. The drawback here is the computational and memory overhead because of the additional data.

2.1 Features of Dynamic Shortest-Path Algorithms

We want to have a closer view on the properties and features of dynamic sp-algorithms. Common to all algorithms is that they want to update only the affected part of a network. The directly affected nodes are in the subtree of the shortest-path tree below the end node of the changed edge. These nodes must be updated at least.

A distinctive feature of dynamic sp-algorithms is the number of nodes for which the optimal route is calculated. Algorithms solving the 1:1-problem calculate an optimal route between one start node and one destination node. In the 1:n-problem we have one start node and an optimal route to nodes in the graph will be calculated. The m:n-problem handles several start nodes and calculates shortest paths to all other nodes but in this paper we focus on the 1:1-problem and 1:n-problem.

Another interesting feature is the number of changes which can be handled at a time. Some algorithms can take only single changes into account and other algorithms are not limited in the number of changes. Multiple changes can be split up into homogeneous changes which include only cost increases or cost decreases but not both and heterogeneous changes which include both types of changes. The dynamic algorithms can also be distinguished by the required input. Some algorithms are able to calculate a shortest-path tree on their own and there are also algorithms which need a pre-calculated shortest-path tree.

2.2 Comparison of Dynamic Shortest-Path Algorithms

We describe the functionality and methodology of dynamic sp-algorithms for the 1:1-problem and 1:n-problem. All algorithms maintain a sorted list of nodes (priority queue) in order to identify the next node to process during the calculation.

Ramalingam provides two different algorithms. The first algorithm [2] solves the 1:n-problem and is split up into two methods, one for handling cost increases and one for handling cost decreases and there are only single changes possible. The algorithm maintains a shortest-path graph which is a shortest-path tree including all other edges which are part of a shortest-path. The shortest-path graph is needed as an input for this algorithm.

In the case of a cost decrease the algorithm inserts the end node of the changed edge into the priority queue if the cost decrease improves the distance of the end node. The sort key in the priority queue relates to the inserted node and not to the start node. In the case of a cost increase it proceeds in two phases. The first phase computes the set of all affected nodes and updates the shortest-path graph. The second phase is a Dijkstra-like calculation, but the priority queue is initialized with all affected nodes having a non-affected predecessor node.

The second algorithm of Ramalingam is called *DynamicSWSF-FP* [2] and bases on viewing the shortest-path problem as a special grammar problem. Ramalingam focuses on the dynamic grammar problem. The algorithm can handle cost increases, cost decreases, and heterogeneous sets of changes. It solves the 1:n-problem and needs no precalculated input. For each node, two distance values are stored, the distance itself and a one-step-look-ahead value called rhs. This rhs value is based on the distance values of the predecessors and is therefore one-step ahead the distance.

The priority queue sorts the nodes by using the minimum of the distance and the rhs value as key. During the calculation the node with the actual minimum key in the priority queue is scanned and its values are compared. If the rhs value is lower than the distance, then the distance is set to the rhs value. Otherwise if the distance is lower than the rhs value, then the distance is resetted to infinity. Afterwards the rhs values of all successors are updated. If the rhs value differs to the distance value of a node then it will be inserted into the priority queue. In the case of a cost change the rhs value of the end node will be updated and checked to be inserted into the priority queue. Afterwards it will start the calculation phase.

The algorithm of *Frigioni* [4] solves the 1:n-problem and needs a calculated shortest-path tree as input. It can handle only single cost changes and is split up into one method for handling a cost increase and one method for handling a cost decrease. The algorithm maintains values called backward and forward level for each neighbour of a node. These values provide information about a shortest-path from the start node to this node passing the neighbour. Another feature of this algorithm is the assignment of each edge to one of its nodes. So each node owns a subset of its incoming and outgoing edges. The calculation phase is based on Dijkstra but the number of scanned successor edges can be limited using the ownership and the levels.

In the case of a cost decrease the end node of the changed edge will be inserted into the priority queue. The calculation phase takes the backward level into account indicating that a decrement will also affect the end node of an edge. In the case of a cost increase the algorithm will proceed in two phases. In the first phase the algorithm marks all nodes with a colour schema. Nodes with no change are white, nodes with a distance change are red and nodes which only change their parent are pink. Afterwards all red nodes having a non-red predecessor are inserted into the priority queue. In the calculation phase the ownership and the forward level information is taken into account.

Demetrescu extends the first Ramalingam and the Frigioni algorithm and presents an own approach [6]. The algorithms can handle single cost increases or decreases and solve the 1:n problem. They need to have a previously calculated shortest-path tree as input. He extends the known algorithms for the use of negative edge weights. In a preprocessing phase the edge costs are transformed into non-negative edge costs. His own algorithm is split up into two methods, one for cost increases and one for cost decreases.

In the case of a cost increase it traverses the affected subtree and inserts these nodes into the priority queue. The sort key for each node is the minimum of the increased value and the possible increase of the node distance using a predecessor node outside the subtree. The Dijkstra-like calculation will calculate the minimal distance variation for all nodes in the subtree with respect to the previously calculated shortest-path tree. In the case of a cost decrease the algorithm inserts the end node of the changed edge into a priority queue and starts a Dijkstra-like calculation phase. The keys used in the priority queue are negative since they indicate the distance variation of each node.

Koenig [5] has developed an extension of the DynamicSWSF-FP algorithm. The algorithm is called *Lifelong Planning A** (LPA*) and

uses a heuristic value to guide the route search towards the destination. The algorithm solves the 1:1-problem and needs no previously calculated input. It can handle cost decreases and cost increases as well as heterogeneous sets of changes.

The LPA* also maintains a distance value and a one-step-look-ahead for each node. It stops the calculation if the distance between start and destination cannot be improved in the next calculation steps. The sort key of the priority queue consists of two values whereby the first value takes the heuristic value into account. The priority queue is sorted lexicographically. The handling of cost changes and the calculation phase is common to the DynamicSWSF-FP algorithm.

Narvaez presents a framework of algorithms [3] which includes the static and dynamic algorithms of Dijkstra, BFM and D'Esopo-Pape. In here we refer to the Dijkstra variant. This algorithm handles cost increases, cost decreases, and homogeneous sets of cost changes. It solves the 1:n-problem and needs no pre-calculated input.

In the case of a cost increase the algorithm will traverse the affected subtree and will directly increase the distance of all affected nodes by the same amount. Then the algorithm searches for edges whose end node is inside this subtree. If such an edge improves the distance, then the end node will be inserted into the priority queue. In the case of a cost decrease the algorithm also traverses the affected subtree and decreases the distance of all affected nodes by the same amount. Afterwards the algorithm searches for edges whose start node is inside this subtree. If such an edge improves the distance of the end node, then the algorithm will insert the end node into the priority queue. The following calculation phase is Dijkstra-like.

The algorithm of *Baselau* will be published in [7] and is called DynamicA*. It extends the Narvaez algorithm using a heuristic value so that the algorithm solves the 1:1 problem and can handle cost increases, cost decreases, and heterogeneous sets of changes. It needs no pre-calculated input. The initialization phase works like the phase in the Narvaez algorithm except that all affected nodes will also be inserted into the priority queue in order to hanlde heterogeneous cost changes correctly. The sort key takes the heuristic value into account and the calculation phase equals an A* algorithm.

3 Summary and Outlook

The comparison made in the previous sections is summarized in table 1. We focussed on the functionality and methodology of a subset of

Table 1. Comparison of the dynamic shortest-path algorithms

Algorithm	Problem solved	Multiple changes	Calculated input needed	Key properties
Ramalingam	1:n	no	yes	shortest-path graph
DynamicSWSF-FP	1:n	yes	no	one-step-look-ahead
Frigioni	1:n	no	yes	level and owner
Demetrescu	1:n	no	yes	negative edge weights
LPA*	1:1	yes	no	one-step-look-ahead uses heuristic
Narvaez	1:n	yes	no	directly change proceeding
Dynamic A*	1:1	yes	no	uses heuristic directly change proceeding

dynamic sp-algorithms solving the 1:1-problem or 1:n-problem. All algorithms try to find the affected subtree(s) of one or more changes and to set up a new priority queue for the re-calculation of the node distances. The algorithms use additional information in order to reduce the number scanned nodes or to sort the priority queue in a special way. Since we had no look at the computational complexity of the algorithms we would like to refer to [2] and [4]. Experimental studies of dynamic sp-algorithms solving the 1:n-problem can be found in [4] and [6] and for the 1:1-problem we refer to [7].

References

1. Ravindra Ahuja et al. (1993) Network Flows. Prentice-Hall
2. Ganesan Ramalingam (1996) Bounded Incremental Computation. Lecture Notes in Computer Science, Number 1089, Springer Verlag
3. Paolo Narvaez (2000) Routing Reconfiguration in IP Networks. PhD thesis, Massachusetts Institute of Technology
4. Daniele Frigioni et al. (2000) Fully dynamic algorithms for maintaining shortest paths trees. Journal of Algorithms 34:251–281
5. Sven Koenig et al. (2001) Lifelong planning A*. Technical Report GITCOGSCI-2002/2, College of Computing, Georgia Institute of Technology, Atlanta (Georgia)
6. Camil Demetrescu (2001) Fully Dynamic Algorithms for Path Problems on Directed Graphs. PhD thesis, University of Rome "La Sapienza"
7. Sven Baselau (to be published) A* - Routensuche in zeitlich variablen Netzen (working title). PhD thesis, Universität Hildesheim

Solving Railway Track Allocation Problems*

Ralf Borndörfer and Thomas Schlechte

Konrad-Zuse-Zentrum für Informationstechnik Berlin, Takustr.7, 14195 Berlin, Germany. {borndoerfer,schlechte}@zib.de

Summary. The *optimal track allocation problem* (OPTRA), also known as the train routing problem or the train timetabling problem, is to find, in a given railway network, a conflict-free set of train routes of maximum value. We propose a novel integer programming formulation for this problem that is based on additional 'configuration' variables. Its LP-relaxation can be solved in polynomial time. These results are the theoretical basis for a column generation algorithm to solve large-scale track allocation problems. Computational results for the Hanover-Kassel-Fulda area of the German long distance railway network involving up to 570 trains are reported.

1 Introduction

Routing a maximum number of trains in a conflict-free way through a track network is one of the basic scheduling problems for a railway company. The problem has received growing attention in the operations research literature recently, see, e.g., [2], [5], [6], [1], [4], [3]. All of these articles model the track allocation problem in terms of a multi-commodity flow of trains in an appropriate time expanded graph, ruling out conflicts by additional packing constraints.

The main problem with this approach is that the resulting integer programs become notoriously difficult already for small problem sizes. This is due to an enormous number of (weak) packing constraints in the model. The purpose of this article is to propose a new formulation of the 'extended' type, that handles conflicts not in terms of constraints, but in terms of additional variables. Our formulation has a constant number of rows, is amenable to standard column generation techniques, and therefore suited for large-scale computation.

* Supported by the Federal Ministry of Economics and Technology (BMWi), grant 19M4031A.

2 The Optimal Track Allocation Problem

The optimal track allocation problem can be formally described in terms of a digraph $D = (V, A)$. Its nodes represent arrivals and departures of trains at a set S of stations at discrete times $T \subseteq \mathbb{Z}$, its arcs model runs of trains between stations. Denote by $s(v) \in S$ the station associated with departure or arrival $v \in V$, and by $t(v) \in T$ the time of this event; we assume $t(u) < t(v)$ for each arc $uv \in A$ such that D is acyclic. Denote by $J = \{s(u)s(v) : uv \in A\}$ the set of all railway tracks. We are further given a set I of requests to route trains through D. More precisely, train $i \in I$ can be routed on a path through some suitably defined subdigraph $D_i = (V_i, A_i) \subseteq D$ from a starting point $s_i \in V_i$ to a terminal point $t_i \in V_i$; let P_i be the set of all routes for train $i \in I$, and $P = \bigcup_{i \in I} P_i$ the set of all train routes (taking the disjoint union). We say that an arc $uv \in A$ *blocks* the underlying track $s(u)s(v)$ during the time interval $[t(u), t(v) - 1]$, that two arcs $a, b \in A$ are *in conflict* if their respective blocking intervals overlap, and that two routes $p, q \in P$ are in conflict if any of their arcs are in conflict. A *track allocation* or timetable is a set of conflict-free routes, at most one for each train. Given arc weights w_a, $a \in A$, the weight of route $p \in P$ is $w_p = \sum_{a \in p} w_a$, and the weight of a track allocation $X \subseteq P$ is $w(X) = \sum_{p \in X} w_p$. The *optimal track allocation problem* (OPTRA) is to find a track allocation of maximum weight.

We refer the reader to the articles [5], [6], [1] an [4]. for discussions how this basic model can be set up to deal with various technical and operational requirements such as preferences for departure, arrival, and travel times, train driving dynamics, single and double tracks, zero-level crossings, station capacities, headways, dwell and turnover times, routing corridors, correspondences, complementarities, and synergies between trains etc.

OPTRA is $\mathcal{NP}$-hard [6]. It can be seen as a multi-commodity flow problem with additional packing constraints, which can be modeled in terms of inequalities [5], [6], [1] [4] and [3]. We propose here an alternative formulation that is based on arc 'configurations', i.e., sets of arcs on the same underlying track that are mutually not in conflict. Formally, let $A_{st} = \{uv \in A : s(u)s(v) = st\}$ be the set of all arcs associated with some track $st \in J$; a *configuration* for this track st is a set of arcs $q \subseteq A_{st}$ that are mutually conflict-free. Let Q_j denote the set of all configuration associated with track $j \in J$, and $Q = \bigcup_{j \in J} Q_j$ the set of all configurations.

Introducing 0/1-variables x_p, $p \in P$, and y_q, $q \in Q$, OPTRA can be stated as the following integer program.

$$
\begin{aligned}
(\text{PCP}) \quad (\text{i}) \quad & \max \sum_{p \in P} w_p x_p & & \\
(\text{ii}) \quad & \sum_{p \in P_i} x_p \leq 1, & & \forall i \in I \\
(\text{iii}) \quad & \sum_{q \in Q_j} y_q \leq 1, & & \forall j \in J \\
(\text{iv}) \quad & \sum_{a \in p \in P} x_p - \sum_{a \in q \in Q} y_q \leq 0, & & \forall a \in A \\
(\text{v}) \quad & x_p, y_q \geq 0, & & \forall p \in P,\ q \in Q \\
(\text{vi}) \quad & x_p, y_q \in \mathbb{Z}, & & \forall p \in P,\ q \in Q.
\end{aligned}
$$

The objective PCP (i) maximizes the weight of the track allocation. Constraints (ii) state that a train can run on at most one route, constraints (iii) allow at most one configuration for each track. Inequalities (iv) link train routes and track configurations to guarantee a conflict-free allocation, (v) and (vi) are the non-negativity and integrality constraints. Note that the upper bounds $x_p \leq 1$, $p \in P$, and $y_q \leq 1$, $q \in Q$, are redundant.

3 Column Generation

Consider the LP-relaxation PLP of PCP, i.e., PLP = PCP (i)–(v); it can be solved by column generation. In fact, it will turn out that the pricing problems for both the route and the configuration variables can be solved in polynomial time by computing longest paths in appropriate acyclic graphs. To see this, consider the dual DLP of PLP.

$$
\begin{aligned}
(\text{DLP}) \quad (\text{i}) \quad & \min \sum_{j \in J} \pi_j + \sum_{i \in I} \gamma_i & & \\
(\text{ii}) \quad & \gamma_i + \textstyle\sum_{a \in p} \lambda_a \geq w_p & & \forall p \in P_i,\ i \in I \\
(\text{iii}) \quad & \pi_j - \textstyle\sum_{a \in q} \lambda_a \geq 0 & & \forall q \in Q_j,\ j \in J \\
(\text{iv}) \quad & \gamma_i, \pi_j, \lambda_a \geq 0 & & \forall i \in I,\ j \in J,\ a \in A.
\end{aligned}
$$

Here, γ_i, $i \in I$, π_j, $j \in J$, and λ_a, $a \in A$, are the dual variables associated with constraints PLP (i), (ii), and (iii), respectively. The pricing problem for a route $p \in P_i$ for train $i \in I$ is

$$
\exists\, p \in P_i : \gamma_i + \sum_{a \in p} \lambda_a < w_p \iff \sum_{a \in p} (w_a - \lambda_a) > \gamma_i.
$$

This is the same as finding a longest $s_i t_i$-path in D_i w.r.t. arc weights $w_a - \lambda_a$; as D_i is acyclic, this problem can be solved in polynomial time.

The pricing problem for a configuration $q \in Q_j$ for track $j \in J$ is

$$\exists\, q \in Q_j : \pi_j - \sum_{a \in q} \lambda_a < 0 \iff \sum_{a \in q} \lambda_a > \pi_j.$$

Let $j = st$ and consider the construction illustrated in Figure 1. Denote by $A_{st} = \{uv \in A : s(u)s(v) = st\}$ the set of arcs that run on track st and by $L_{st} := \{u : uv \in A_{st}\}$ and $R_{st} := \{v : uv \in A_{st}\}$ the associated set of departure and arrival nodes; note that all arcs in A_{st} go from L_{st} to R_{st}. Let $\overline{A}_{st} := \{vu : t(v) \leq t(u), v \in R_{st}, u \in L_{st}\}$ be a set of 'return' arcs that go in the opposite direction. It is easy to see that $D_{st} = (L_{st} \cup R_{st}, A_{st} \cup \overline{A}_{st})$ is bipartite and acyclic, and that $L_{st}R_{st}$-paths $a_1, \overline{a}_1, \ldots, \overline{a}_{k-1}, a_k$ in D_{st} and configurations $a_1, \ldots, a_k$ in Q_{st} are in 1-1 correspondence. Using arc weights λ_a, $a \in A_{st}$, and 0, $a \in \overline{A}_{st}$, pricing configurations in Q_{st} is equivalent to finding a longest $L_{st}R_{st}$-path in D_{st}. As D_{st} is acyclic, this is polynomial. It follows

Theorem 1. PLP *can be solved in polynomial time.*

In practice, tailing-off prevents the straightforward solution of PLP to optimality. However, the path lengths $\max_{p \in P_i} \sum_{a \in p}(w_a - \lambda_a)$ and $\max_{q \in Q_j} \sum_{a \in q} \lambda_a$ yield the following bound $\beta = \beta(\gamma, \pi, \lambda)$.

Lemma 1. *Let* $\gamma, \pi, \lambda \geq 0$ *be dual variables*[2] *for* PLP *and* v(PLP) *the optimum of* PLP. *Let* $\eta_i := \max_{p \in P_i} \sum_{a \in p}(w_a - \lambda_a) - \gamma_i$, $i \in I$, *and* $\theta_j := \max_{q \in Q_j} \sum_{a \in q} \lambda_a - \pi_j$, $j \in J$. *Then:*

$$v(\text{PLP}) \leq \sum_{i \in I} \max\{\gamma_i + \eta_i, 0\} + \sum_{j \in J} \max\{\pi_j + \theta_j, 0\} =: \beta(\gamma, \pi, \lambda).$$

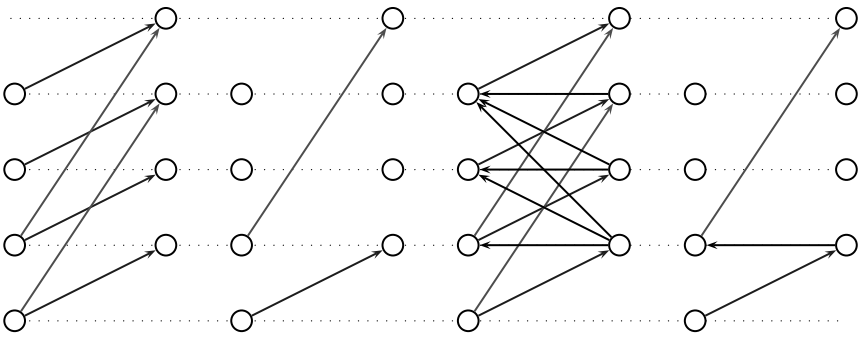


Fig. 1. Arc configurations on a track. From left to right: train routing digraph, conflict-free configuration, configuration routing digraph, and LR-path

[2] Note that these will be infeasible during column generation.

4 Computational Results

We have implemented a column generation algorithm for the PCP along the lines of the preceding sections. We have used this code to solve three large-scale railway track allocation problems for the Hannover-Kassel-Fulda area of the German long-distance railway network involving 146, 250, and 570 trains, see Table 1. The instances are based on a common macroscopic infrastructure model with 37 stations and 120 tracks, 6 different train types (ICE, IC, RE, RB, S, ICG), and 4320 headway times, see Figure 2 for an illustration and [1] for a more detailed discussion.

Figure 3 illustrates the solution of the LP-relaxation PLP for the two large scenarios 2 and 3. It can be seen that the upper bound $\beta(\gamma, \pi, \lambda)$ and the optimal value $v(RPLP)$ of the restricted master-LP converge, i.e., we can indeed solve these LPs close to optimality. This provides a good starting point to compute high-quality integer solutions using standard rounding heuristics, see columns IP and gap in Table 1. All computations were made single-threaded on a Dell Precision 650 PC with 2GB of main memory and a dual Intel Xeon 3.8 GHz CPU running SUSE Linux 10.1. The reduced master-LPs were solved with CPLEX 10.0 using the barrier or dual simplex method, depending on the column generation progress.

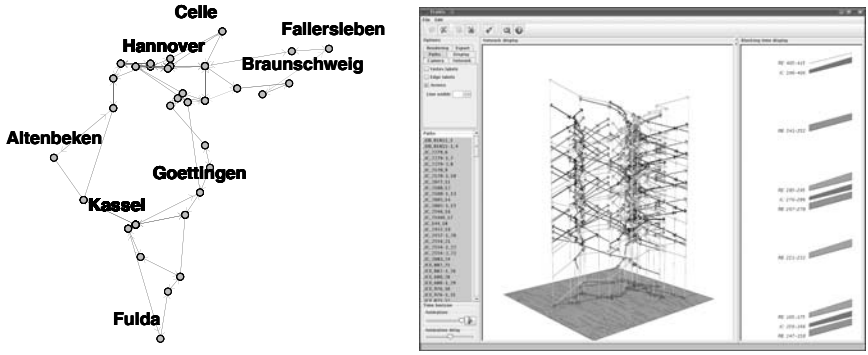


Fig. 2. Infrastructure network (left), visualization of an allocation (right)

Table 1. Solving large-scale railway track allocation problems

no	$\|I\|$	*rows*	*cols*[3]	*iter*	β	$v(RPLP)$	IP	gap in %	time in sec.
1	146	6034	120366	162	93418	93381	93371	0.05	4439
2	250	12461	213218	168	148101	147375	147375	0.75	39406
3	570	11112	250550	148	245278	239772	234538	4.58	59910

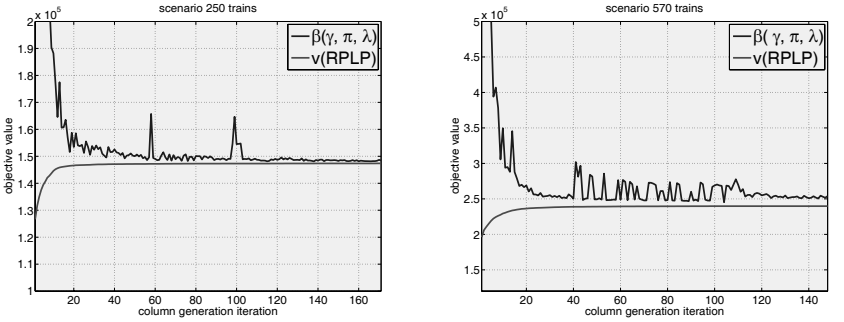


Fig. 3. Solving the LP-relaxations of scenario 2 (left) and 3 (right) by column generation

References

1. R. Borndörfer, M. Grötschel, S. Lukac, K. Mitusch, T. Schlechte, S. Schultz, and A. Tanner. An auctioning approach to railway slot allocation. *Competition and Regulation in Network Industries*, 1(2):163–196, 2006.
2. U. Brännlund, P. Lindberg, A. Nou, and J.-E. Nilsson. Railway timetabling using langangian relaxation. *Transportation Science*, 32(4):358–369, 1998.
3. V. Cacchiani. *Models and Algorithms for Combinatorial Optimization Problems arising in Railway Applications.* PhD thesis, DEIS, Bologna, 2007.
4. V. Cacchiani, A. Caprara, and P. Toth. A column generation approach to traintimetabling on a corridor. *4OR*, 2007. To appear.
5. A. Caprara, M. Fischetti, P. L. Guida, M. Monaci, G. Sacco, and P. Toth. Solution of real-world train timetabling problems. In *HICSS 34*. IEEE Computer Society Press, 2001.
6. A. Caprara, M. Fischetti, and P. Toth. Modeling and solving the train timetabling problem. *Operations Research*, 50(5):851–861, 2002.

[3] *cols* is the max. number of columns in main memory during column generation.

On a Class of Interval Data Minmax Regret CO Problems

Alfredo Candia-Véjar and Eduardo Álvarez-Miranda

Departamento de Modelación y Gestión Industrial; Universidad de Talca, Curicó, Chile. acandia@utalca.cl, edam@alumnos.utalca.cl

Summary. Some remarks about the Kasperski and Zielinski approximation algorithm for a class of interval data minmax regret combinatorial optimization problems (Algorithm *K&Z*) are presented. These remarks help to give a better understanding of both the design of the algorithm and its possible applications.

Key words: Approximation algorithm, minmax regret, interval data.

1 Introduction

Interval data minmax regret combinatorial optimization problems (IDMRCO) is a class of *Combinatorial Optimization problems* (CO) for which a particular approach is used to model uncertainty associated to data. This approach contains two specific ingredients: a *minmax regret* objective function and intervals to represent the uncertain parameters in the objective function. Uncertainty can be structured through the concept of scenario corresponding to an assignment of plausible values to model parameters. Two ways of describing the set of all possible scenarios are typically considered: *interval data*, where each numerical parameter can take any value between a lower and an upper bound; and *discrete scenarios,* where the scenario set is described explicitely.

The book by Kouvelis and Yu [6] presenting practical motivations for studying minmax regret CO problems in the framework of CO problems with uncertainty associated to data, also discusses the existence of other approaches considering uncertainty (the probabilistic approach, for example), and as well gives a survey of the developments in this area until 1996. IDMRCO problems such as minimum spanning tree, shortest path and assignment problem have been analyzed. It is known

these problems in their classical versions can be solved in polynomial time; however, their interval data minmax regret versions are NP-Hard; [2] and [10]. Several works have proposed both exact algorithms (for example, see [7] and [9]) and heuristics for these problems ([8] and [3]).

The study of efficient approximation algorithms for NP-Hard IDMRCO problems is an important challenge. The first published work on this topic is an approximation algorithm with a *performance ratio* of 2, which could be applied to a wide class of IDMRCO problems [5]. In that paper, the approximation algorithm proposed is called algorithm *AM*, but from now on we refer to this algorithm as algorithm *K&Z*. This recent work is important by two reasons; it proves the existence of polynomial time approximation algorithms with a constant performance ratio for this important class of IDMRCO problems and it also provides good feasible solutions, which should help in the design of more sophisticated approaches to these problems. In fact, in [3], algorithm *K&Z* was implemented and computational experiments compared solutions from a heuristic proposed for the minmax regret spanning arborescence problem with the solutions provided by algorithm *K&Z*. Also, in [8], algorithm *K&Z* was applied in the development of heuristics and preprocessing techniques for the interval data minmax regret travelling salesman problem.

Our paper presents an analysis of the algorithm *K&Z*, making clear the significance of considering the scenario S in which the costs of the elements are the midpoints of their corresponding cost intervals.

2 Problem Formulation and Algorithm *K&Z*

For simplicity we will use almost the same notation given in [5]. Let $E = \{e_1, ..., e_n\}$ be a finite set, $\mid E \mid = n$, and $\Phi \subseteq 2^E$ be a set of subsets of E. Set Φ is called the *feasible solutions* set. For every element $e \in E$, there is given an interval $I_e = [c_e^-, c_e^+]$, which expresses a range of possible values for the cost. A *scenario* is a vector $S = (c_e^s)_{e \in E}$ that represents a particular assignment of costs c_e^s to elements $e \in E$ and $\Gamma = \otimes_{e \in E} I_e$ is the Cartesian product of the corresponding intervals I_e, $e \in E$. For a given solution $X \in \Phi$, its cost under a fixed scenario $S \in \Gamma$ is defined as follows:

$$F(X, S) = \sum_{e \in X} c_e^s.$$

Furthermore, $F^*(S)$ will denote the value of the cost of the optimal solution under scenario $S \in \Gamma$,

$$F^*(S) = \min_{X \in \Phi} F(X, S). \quad (1)$$

When S is fixed, the *classical combinatorial optimization problem* it is obtained. An important hypothesis assumed in this paper is that there is a polynomial time algorithm which outputs an optimal solution for problem (1) for a fixed scenario S. Now, the *maximal regret* for $X \in \Phi$ is defined as follows:

$$Z(X) = \max_{S \in \Gamma} \{F(X, S) - F^*(S)\} \quad (2)$$

Scenario S which maximizes the right-hand side of (2) is called the *worst-case scenario* for X.

The minmax regret combinatorial optimization problem **P** associated with problem (1) is to find a feasible solution for which the maximal regret is minimal:

$$\mathbf{P} : \min_{X \in \Phi} Z(X).$$

Note that problem (1) is a special case of problem **P** if set Γ consists of a single scenario.

It is known that the worst case scenario for a given solution $X \in \Phi$ can be characterized in the following way (see [1]).

Proposition 1. *Given a solution $X \in \Phi$, the worst case scenario S^X for X is the one where elements $e \in X$ have costs c_e^+ and all the other elements have costs c_e^-; i.e., $c_e^{S^X} = c_e^+$ if $e \in X$, and $c_e^{S^X} = c_e^-$ if $e \in E \backslash X$.*

Thus, the maximal regret of a given solution $X \in \Phi$ can be formulated as follows:

$$Z(X) = F(X, S^X) - F^*(S^X).$$

Algorithm *K&Z* for solving **P** is presented now. Let $AOpt(S)$ denotes a polynomial algorithm that outputs an optimal solution for the underlying classical combinatorial optimization problem for a fixed scenario S (see the problem (1)).

Algorithm ***K&Z***
for all $e \in E$ do
$c_e^S \longleftarrow \frac{1}{2}(c_e^- + c_e^+)$;
end for
$M \longleftarrow AOpt(S)$;
return M;

The main results obtained in [5] are presented now.

Proposition 2. *Let M be the solution constructed by algorithm* K&Z. *Then for all $X \in \Phi$ it holds that $Z(M) \leq 2Z(X)$.*

Theorem 1. *The ratio performance of algorithm* K&Z *is at most 2.*

3 Remarks on Algorithm *K&Z*

Our first remark is centrated on the first step of algorithm *K&Z*; that is, on the definition of the scenario S fixing each arc cost as the *midpoint* of $[c_e^-, c_e^+]$.

Lemma 1. *Proposition 2 remains valid if we replace the number $\frac{1}{2}$ by any positive real number p when the scenario S is defined in the algorithm* K&Z.

Proof. It is enough to note that if $\mathcal{P} = argmin\{\sum_{e \in X} pc_e : X \in \Phi\}$, $p > 0$ then $\mathcal{P}$ is unique for any $p > 0$.

Our result shows that this property essentially comes from the sum of the extreme values of each interval cost. Note that scenarios defined by the values of p might not belong to Γ; in fact, it is clear that $c_e^s \longleftarrow p(c_e^+ + c_e^-)$ defines a valid scenario only if $p \in [\frac{c_e^-}{c_e^+ + c_e^-}, \frac{c_e^+}{c_e^+ + c_e^-}]$.

Our second remark (Lemma 2), analyzes the existence of multiple solutions for $AOpt(S)$.

In Theorem 1 it is proved that the performance ratio of algorithm *K&Z* is at most 2. Therefore, if Y is an output for $AOpt(S)$ in algorithm *K&Z*, then $Y \in \Phi$, $Z^* = \min_{X \in \Phi} Z(X) \leq Z(Y)$ and $Z(Y) \leq 2Z^*$. So, the optimal value Z^* for $\mathbf{P}$ satisfies

$$\frac{Z(Y)}{2} \leq Z^* \leq Z(Y).$$

Furthermore, since the problem $F^*(S) = \min_{X \in \Phi} F(X, S)$ could have multiple optimal solutions $Y_i, i = 1, ..., k$, an algorithm to obtain the k-best solutions for a combinatorial optimization problem could be applied to obtain k feasible solutions for $\mathbf{P}$ and consequently k intervals for Z^*, given by $I_i = [\frac{Z(Y_i)}{2}, Z(Y_i)]$. Then, from this information, it is easy to obtain a new interval for Z as detailed in the following result.

Lemma 2. *If the resolution of problem $\boldsymbol{P}$ considers a scenario given by $\{c_e^s\} = \{p(c_e^+ + c_e^-\}$, $p > 0$, and then an algorithm is applied to obtain the k optimal solutions for the underlying problem $F^*(s)$; then $Z^* \in [\alpha, \beta]$, where: $\alpha = \max_{i=1,...,k}\{\frac{Z(Y_i)}{2}\}$ and $\beta = \min_{i=1,....,k}\{Z(Y_i)\}$.*

According Lemma 1 and Lemma 2 the following generalization of algorithm *K&Z* is valid.

Algorithm ***AP***

(Algorithm for the sum of the intervals extreme values)
Input: $E = \{e_1, ..., e_n\}$, $[c_e^-, c_e^+]$ for each $e \in E$, $p > 0$
Output: P, a feasible solution, and I, an interval containing Z^*
for all $e \in E$ **do**
$c_e^s \longleftarrow p(c_e^+ + c_e^-)$;
end for
$\mathcal{P} \longleftarrow AOpt(S)$;
for all $Y \in \mathcal{P}$ **do**
$z^Y \longleftarrow Z(Y)$;
end for
$P \longleftarrow \arg\min\{z^Y : Y \in \mathcal{P}\}$;
$I \longleftarrow [\alpha, \beta]$
return P, I;

According to above results, we have the following theorem.

Theorem 2. *The performance ratio of algorithm* AP *is at most 2.*

A final open question about this issue refers to knowing if the bound 2 for the ratio $\frac{Z(P)}{Z^*}$ is tight.

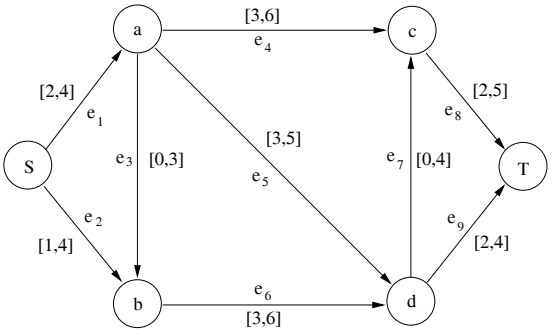


Fig. 1. The graph illustrates an example where $AOpt(S)$ returns two different solutions

Several algorithms have been designed for finding the best k solutions for CO problem, especially, for the shortest paths in a network (see [4]). Note that solving **P** using algorithm *K&Z* implies solving only

two classic problems $F^*(S)$; one to find Y and one for finding $Z(Y)$. To solve $\mathbf{P}$ when $F^*(S)$ has k optimal solutions takes time $O(f(k,n))$ $+ k \cdot O(g(n))$, where $f(k,n)$ is the time for finding the best k solutions for the classic CO problem, and $g(n)$ is the time for finding $Z(Y)$.

Example 1. Consider the graph described in Figure 1 as an instance of the IDMR Shortest Path Problem . Applying algorithm AP, we have $\mathcal{P} = \{\{e_1, e_5, e_9\}, \{e_2, e_6, e_9\}\} = \{Y_1, Y_2\}$. Then, $Z(Y_1) = 13 - 8 = 5$ and $Z(Y_2) = 14 - 7 = 7$; therefore, it results $Z^* \in [3.5, 5]$.

Acknowledgement. This research has been supported by the European Alfa Project II-0321-FA:Engineering Systems for Preparing and Making Decisions Under Multiple Criteria (SISTING) http://www.sisting.net.

References

1. I. Averbakh, On the complexity of a class of combinatorial optimization problems with uncertainty, Mathematical Programming, Ser.A 90 (2001) 263-272.
2. I. Averbakh and Lebedev, Interval data min-max regret network optimization problems, Discrete Applied Mathematics 138 (2004) 289-301.
3. E. Conde and A. Candia, Minmax regret spanning arborescences under uncertain costs, European Journal of Operational Research 182 (2007) 561-577.
4. D. Eppstein, Finding the k Shortest Paths, SIAM J. Comput. 28(2) (1998) 652-673.
5. A. Kasperski and P. Zielinski, An approximation algorithm for interval data minmax regret combinatorial optimization problems, Information Processing Letters 97 (2006) 177-180.
6. P. Kouvelis and G. Yu, Robust discrete optimization and Its Applications, Kluwer Academic Publishers, Boston, 1997.
7. R. Montemanni, A Benders decomposition approach for the robust spanning tree problem with interval data, European Journal of Operational Research 174(3) (2006) 1479-1490.
8. R. Montemanni, J. Barta and L. M. Gambardella, Heuristic and preprocessing techniques for the robust traveling salesman problem with interval data, Technical Report IDSIA-01-06.
9. H. Yaman, O.E. Karasan and M.C. Pinar, The robust spanning tree problem with interval data, Operations Research Letters 29 (2001) 31-40.
10. P. Zielinski, The computational complexity of the relative robust shortest path probem with interval data, European Journal of Operational Research 158 (2004) 570-576.

A Benders Decomposition for Hub Location Problems Arising in Public Transport

Shahin Gelareh[1] and Stefan Nickel[2]

[1] Fraunhofer Institut für Techno-und Wirtscaftsmathematik(ITWM), D 67663 Kaiserslautern, Germany. gelareh@itwm.fhg.de
[2] Universität des Saarlandes, Germany. s.nickel@orl.uni-saarland.de

1 Introduction

In the last two decades, among the large amount of literature available on Hub Location Problems (HLP), applications in Public Transport (PT) have received less attentions compared to other fields, for example telecommunications. The first mathematical model for HLPs is proposed by O'Kelly [5] in 1987. In a HLP network, the flow originated from an origin i and destinated to node j is not shipped directly, rather, it is sent via some selected intermediate nodes (called *hub nodes*) and maybe intermediate edges (called *hub edges*) connecting these hubs. The sub-network composed of these hub facilities is known as *hub-level network*. The remaining nodes and edges are called *spoke nodes* or *spoke edges* of the *spoke-level* network. For applications in public transport planning, the hub level network consists of special types of transportation facilities which may be fast-lines, etc. Fig. 1 depicts a simple hub location model applied to public transport planning.

The hubs can have three main functionalities, namely: *(i) Consolidation (concentration)* of the flows which they receive, in order to have a larger flow and making use of economy of scale, *(ii) Switching(transfer)* stations, which allow the flows to be re-directed and *(iii) Distribution(decomposition)* of large flows into smaller ones.

Among many reviews and contributions we refer readers to [1] and references therein. For solution approaches to HLPs we refer to [2]. In the scope of public transport we refer to [4] where the first mathematical model for HLPs in public transport is proposed.

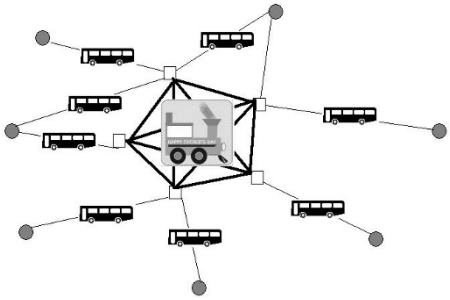


Fig. 1. A typical Public Transport Network

2 Mathematical Formulation

In this section we derive a new mixed integer programming (MIP) model for applying HLP in public transport (HLPPTs). Contrary to the classical assumptions of HLPs, our hub-level network is not necessarily a complete graph. Moreover, considering the nature of public transportation, the triangle inequality does not hold in general. The parameters of the model are as follows:

n : the number of nodes	F_k : hub k setup cost,
c_{kl} : traveling cost from k to l	I_{kl} : hub edge (k,l) setup cost,
W_{ij} : amount of passengers going from i to j	α : hub edge discount factor.

The variables in this model are defined as follows: $x_{ijkl} = 1, i \neq j, k \neq l$ if the optimal path from i to j traverses the hub edge (k,l) and 0, otherwise. Also, $a_{ijk} = 1, j \neq i, k \neq i, j$ if the optimal path from i to j traverses (i,k) and i is not hub and 0, otherwise and $b_{ijk} = 1, j \neq i, k \neq i, j$ if the optimal path from i to j traverses (k,j) and j is not hub and 0, otherwise. In addition, $e_{ij} = 1, i \neq j$ if the optimal path from i to j traverses (i,j) and either i or j is hub and 0, otherwise. For the hub-level variables, $y_{kl} = 1, k < l$ if the hub edge (k,l) is established and 0, otherwise and $h_k = 1$ if k is used as hub 0 otherwise.

The flow cost for a given flow with origin i and destination j is the sum of, (i) the cost of sending flow from i to the first hub node, (ii) the cost of traversing one or more hub edges discounted by the discount factor $0 < \alpha < 1$ and (iii) the cost of connecting the last hub node to the destination. The proposed mathematical formulation turns out to be as follows:

(HLPPT)

$$Min \sum_i \sum_{j\neq i} \sum_k \sum_{l\neq k} \alpha W_{ij} C_{kl} x_{ijkl} + \sum_i \sum_{j\neq i} \sum_{k\neq i,j} W_{ij} C_{ik} a_{ijk} + \sum_i \sum_{j\neq i} \sum_{k\neq i,j} W_{ij} C_{kj} b_{ijk} + \sum_i \sum_{j\neq i} W_{ij} C_{ij} e_{ij} + \sum_k F_k h_k + \sum_k \sum_{l>k} I_{kl} y_{kl} \quad (1)$$

$$s.t. \sum_{l\neq i} x_{ijil} + \sum_{l\neq i,j} a_{ijl} + e_{ij} = 1, \qquad \forall i, j \neq i \quad (2)$$

$$\sum_{l\neq j} x_{ijlj} + \sum_{l\neq i,j} b_{ijl} + e_{ij} = 1, \qquad \forall i, j \neq i \quad (3)$$

$$\sum_{l\neq k,i} x_{ijkl} + b_{ijk} = \sum_{l\neq k,j} x_{ijlk} + a_{ijk}, \qquad \forall i, j \neq i, k \neq i, j \quad (4)$$

$$y_{kl} \leq h_k, \quad y_{kl} \leq h_l, \qquad \forall k, l > k \quad (5)$$

$$x_{ijkl} + x_{ijlk} \leq y_{kl}, \qquad \forall i, j \neq i, k, l > k \quad (6)$$

$$\sum_{l\neq k} x_{kjkl} \leq h_k, \qquad \forall j, k \neq j \quad (7)$$

$$\sum_{k\neq l} x_{ilkl} \leq h_l, \qquad \forall i, l \neq i \quad (8)$$

$$e_{ij} \leq 2 - (h_i + h_j), \qquad \forall i, j \neq i \quad (9)$$

$$a_{ijk} \leq 1 - h_i, \qquad \forall i, j \neq i, k \neq i, j \quad (10)$$

$$b_{ijl} \leq 1 - h_j, \qquad \forall i, j \neq i, l \neq i, j \quad (11)$$

$$a_{ijk} + \sum_{l\neq j,k} x_{ijlk} \leq h_k, \qquad \forall i, j \neq i, k \neq i, j \quad (12)$$

$$b_{ijk} + \sum_{l\neq k,i} x_{ijkl} \leq h_k, \qquad \forall i, j \neq i, k \neq i, j \quad (13)$$

$$e_{ij} + 2x_{ijij} + \sum_{l\neq j,i} x_{ijil} + \sum_{l\neq i,j} x_{ijlj} \leq h_i + h_j, \quad \forall i, j \neq i \quad (14)$$

$$x_{ijkl}, y_{kl}, h_k, a_{ijk}, b_{ijk}, e_{ij} \in \{0, 1\}. \quad (15)$$

3 Benders Decomposition Method for the HLPPT

Following the Bender's algorithm, our HLPPT can be decomposed into a master problem and a subproblem (separable for each i and j to an independent smaller problem), such that the master problem is a re-

laxation of the original problem, where its feasible region is iteratively tightened using the cuts generated by the solution to the sub-problem. To the best of our knowledge, only [2] has proposed a Benders algorithm for an HLP similar to HLPPT, where they assume a complete hub-level network in their model. In our model, the master problem contains the hub-level cost terms of the objective function together with the constraints that only contain hub-level variables, with one additional constraint which ensures the existence of at least one hub edge in every solution. Such an MP does not guaranty a connected hub-level network. Therefore, an alternative MP is used.

Let $G(H, E)$ be a connected and $G_d = (H, A)$ be a directed graph, where $A = \{(i,j),(j,i)|\{i,j\} \in E\}$. Two new graphs $G^0 = (V_0, E_0)$ and $G_d^0 = (V_0, A_0)$ where $V_0 = V \cup \{0\}$, $E_0 = E \cup \{\{0,j\}|j \in V\}$, $A_0 = A \cup \{\{0,j\}|j \in V\}$ are defined.

Let $h = (h_i)_{i\in V} \in \{0,1\}^{|V|}$, $y = (y_u)_{u\in E_0} \in \{0,1\}^{|E_0|}$ two $0-1$ vectors and $z_{ij}^k \geq 0, (i,j) \in A_0$, $k \in V'$ where V' is a subset of V, and z_{ij}^k is a real flow on the arc $(i,j) \in A_0$ having 0 as source and k as destination. $\Gamma(i)$ is considered as the set of edges $u \in E$ having an endpoint as i, $\Gamma^+(i) = \{j|(i,j) \in A_0\}$ and $\Gamma^-(i) = \{j|(j,i) \in A_0\}$, $m = |E|$ and $n = |V|$ [3]. Then MP is defined as follows:

(MP)

$$Min \quad \sum_k F_k h_k + \sum_k \sum_{l>k} I_{kl} y_{kl}$$

$$s.t. \quad \sum_{j\in\Gamma^+(0)} z_{0j}^k - h_k = 0 \qquad \forall k \in V \quad (16)$$

$$\sum_{j\in\Gamma^+(i)} z_{ij}^k - \sum_{j\in\Gamma^-(i)} z_{ji}^k = 0 \qquad \forall i \in V - \{k\}, k \in V \quad (17)$$

$$\sum_{j\in\Gamma^+(k)} z_{kj}^k - \sum_{j\in\Gamma^-(k)} z_{jk}^k + h_k = 0 \qquad \forall k \in V \quad (18)$$

$$z_{ij}^k \leq y_{ij}, \quad z_{ji}^k \leq y_{ij}, \qquad \forall \{i,j\} \in E_0, k \in V \quad (19)$$

$$y_{ij} \leq x_i, \quad y_{ij} \leq x_j, \qquad \forall \{i,j\} \in E \quad (20)$$

$$\sum_{j\in V} y_{0j} = 1 \qquad \forall i,j = 1,\ldots,n \quad (21)$$

$$z_{ij}^k \geq 0 \qquad \forall (i,j) \in A_0, k \in V \quad (22)$$

$$y_{ij} \in \{0,1\}, \{i,j\} \in E_0, h_k \in \{0,1\}, k \in V, \eta \in \mathbb{R}^+ \quad (23)$$

Theorem 1. *All vectors h and y satisfying (16-23) and (24) are associated with connected sub-graphs of G with at least one hub edge.*

$$\sum_{u \in Y} y_u \geq 1 \qquad or \qquad \sum_{i \in H} h_i \geq 2. \tag{24}$$

The dual of sub-problems, for fixed values of y_{kl} and h_k results in a linear program.

Each cut generated for the MP following the Benders algorithm is in form of (25). It can also be generated by aggregation of $n(n-1)$ sub-cuts, where each of them can be directly is achieved as the result of each of $i \times j$-th part of sub-problem corresponding to flow from i to j. Hence, we can have two approaches, one single cut(SC) approach and one multi-cut(MC) approach. In SC, a single cut and in MC, $n(n-1)$ sub-cuts from the solution of the sub-problem are added to the MP.

$$\begin{aligned} &- \sum_{i,j \neq i} ((u_{ij} + v_{ij}) + \sum_{k \neq i,j} (s_{ijk} + w_{ijk}) h_k + p_{ji} h_i + q_{ij} h_j + \sum_{k,l>k} o_{ijkl} y_{kl} \\ &+ \sum_{k \neq i,j} (a_{ijk}(1 - h_i) + b_{ijk}(1 - h_j)) + d_{ij}(h_i + h_j) + e_{ij}(2 - h_i - h_j))) \\ &- \eta \leq 0. \end{aligned} \tag{25}$$

4 Computational Results

In Table 1, computational results comparing CPLEX 9.1 with the single cut and multi-cuts Benders are given. One can observe that the multi-cut approach outperforms the other methods both in terms of problem size and computational time. Thereby, the well-known AP data set with the hub facility setup cost of $F_i := 5000$ and hub edge setup cost of $I_{kl} := d_{kl} \times 500, \forall k, l > k$ has been used. In Table 1, *N.A.* stands for "not available", that is the solver was not able to solve the instance and *E.T.* means that run time exceeds the time limit of 20 hours. These computational results have been obtained on a AMD Opteron 250-2.40 GHz and 1 GB of RAM and with the AP data set.

5 Conclusions

A new mixed integer programming model is proposed for the application of HLPs in public transport planning. The model generalizes the

Table 1. Overall comparison

Instance	*CPLEX 9.1*		*SC*		*MC*	
	T.	*Cpu(sec.)*	*T.*	*Cpu(sec.)*	*T.*	*Cpu(sec.)*
AP5	0.03		8.48		0.56	
AP10	10.39		1500.79		7.75	
AP15	591.57		E.T.		32.27	
AP20	5597.2		E.T.		137.91	
AP25	E.T.		E.T.		695.03	
AP30	N.A.		E.T.		2657.16	
AP35	N.A.		E.T.		10793.37	
AP40	N.A.		E.T.		61032.52	

classical HLPs for applications in public transport by relaxing some of the classical assumptions in HLPs, making the problem more difficult to solve. Due to the combinatorial behavior of HLPs, as the problem size grows, general purpose standard solvers fail to solve larger instances. Our computational results confirms the superiority of the Benders approach. Where, the standard solver CPLEX 9.1 was not able to solve instances of size larger than 20, the multi-cut Benders approach presented here could solve instances of larger size in a significantly smaller amount of time.

References

1. Alumur, S. and Kara, Y.B. (2007) Network hub location problems: the state of the art European Journal of Operational Research (In Press)
2. R.S. de Camargo, Jr., Miranda, G. and Luna, H.P. (2005) Benders decomposition for the uncapacitated multiple allocation hub location problem Computers & OR 32:1429-1450
3. Maculan, N., Plateau, G. and Lisser, A. (2005) Integer linear models with a polynomial number of variables and constraints for some classical combinatorial optimization problems Pesquisa Operacional 23:161 - 168
4. Nickel, S., Schoebel, A. and Sonneborn, T. (2001) Hub location problems in urban traffic networks In Niittymaki and Pursula, editors, Mathematical Methods and Optimization in Transportation Systems, KLUWER academic publishers 39:95-107
5. O'Kelly, M.E. (1987) A quadratic integer program for the location of interacting hub facilities European Journal of Operational Research (32) 393-404

Reliability Models for the Uncapacitated Facility Location Problem with User Preferences

Rodrigo Herrera[1], Jörg Kalcsics[2], and Stefan Nickel[2]

[1] School of Industrial Engineering, Talca University, Chile.
Chair of Econometrics, Dresden University of Technology, Germany
Rodrigo.Herrera@mailbox.tu-dresden.de
[2] Chair in Operation Research, Saarland University, Germany
j.kalcsics@orl.uni-saarland.de, s.nickel@orl.uni-saarland.de

Summary. We consider a fault tolerance version of the Uncapacitated Facility Location Problem with User Preferences. As a consequence, our problem, which we wish to name the Uncapacitated Facility Location Problem with user preferences and q-level of reliability (q-level UFLPP), is much more difficult to solve. A computational study shows the advantages and difficulties of this approach.

Key words: Location, Optimization Modeling, Reliability

1 Introduction

In this paper we consider a fault tolerance version of the Uncapacitated Facility Location Problem with User Preferences. In this problem, the assignment of clients to facilities is not (necessarily) based on distances or costs, but we are given a preference function for each client and clients are assigned to new facilities in order of preference. The main aim is to assign each client to a primary facility, the most preferable of the new facilities for the client, that will serve it under normal circumstances, as well as to a set of backup facilities that serve it when the primary facility has failed, in order of preference for the client in question. Our problem, which we wish to name the Uncapacitated Facility Location Problem with user preferences and q-level of reliability (q-level UFLPP), is NP- hard in strong sense, as it can be reduced to the Multi Stage Uncapacitated Facility Location Problem (MSUFLP) as the described by [3].

Models considering only clients preferences can be found in [5, 4, 2, 1]. In the case of reliability models for facility location context some approaches in this spirit can be found in [7, 6, 8]. As far as we know our approach is new. The remaining part of this paper is organized as follows. In Section 2 we give a tighter formulation of the problem. Section 3 presents the results of a series of computational experiments to analyze the inherent difficulty of the model. In the last section we present conclusions and future works.

2 General Problem Formulation

Denote $\mathcal{I} = \{1, \ldots, m\}$ the set of customers and $\mathcal{J} = \{1, \ldots, n\}$ the set of potential sites for new facilities. The costs of supplying the whole demand of customer $i \in \mathcal{I}$ from facility $j \in \mathcal{J}$ is defined by $c_{ij} \geq 0$ and for opening a facility at j is $f_j \geq 0$. Let q be the number of facilities that a client i should be assigned to so that under failure of up to $(q-1)$ facilities, the client i can still be serviced. Moreover, each client i has a strict level of preference for each facility $j \in \mathcal{J}$. The problem is to find a subset $\mathcal{L} \subseteq \mathcal{J}$ of facilities in such a way that each client i is assigned to q facilities in order of preference, and that the total cost of opening facilities and connecting clients to open facilities is minimized.

Definition 1. *(Facilities failure probability) We define that each facility $j \in \mathcal{J}$ has a probability of becoming nonoperational equal to $p_{ijl} \in [0, 1]$ with $i \in \mathcal{I}$, $l \in \mathcal{L}$,$|\mathcal{L}| = q$ and $q \leq n$. Therefore, a customer i is assigned to one facility j at level l of reliability with a probability $(1 - p_{ijl}) \prod_{t=1}^{l-1} p_{ijt}$.*

Definition 2. *(Customer's preferences) We say that $j \in \mathcal{J}$ is for customer i the d-th prefered facility and we denoted this by d_{ij}. Moreover, for $j, k \in \mathcal{J}$ and $i \in \mathcal{I}$ we say that k is worse than j, denoted $k <_i j$, if client i prefers facility j over facility k.*

Thus, a customer i can be served by a d-th preferred facility j with decreasing level of preference $1 \leq d \leq n$. Moreover, each customer i can be served by its level-l facility (call it j) if the $(l-1)$ more preferred facilities have failed and j itself has not failed.

Note that not all possible combinations of facilities $\mathcal{L} \subseteq \mathcal{J}$ yield a feasible solution, because each set of facilities for a client i_1 must also allow to define a feasible set of facilities for each other client $i \neq i_1 \in \mathcal{I}$. This idea is summarized in the following theorem.

Theorem 1. *(Global preference order). Let $\mathcal{S}_{i_1} \subseteq \mathcal{J}$ be a set of feasible facilities for client $i_1 \in \mathcal{I}$ for each l-level of reliability and let $j_{i_1,w} \in \mathcal{S}_{i_1}$ be the facility with the worst preference ranking for i_1 and $j_{i_1,p} \in \mathcal{S}_{i_1}$ the most preferred (i.e., $j_{i_1,w} <_{i_1} j$, $\forall j \neq j_{i_1,w} \in \mathcal{S}_{i_1}$ and $j_{i_1,p} >_{i_1} j$, $\forall j \neq j_{i_1,p} \in \mathcal{S}_{i_1}$). Define $\mathcal{T}_{i_1}$ as the set of facilities with worse preference than $j_{i_1,w}$ for the client i_1, $\mathcal{T}_{i_1} = \{j \in J : j <_{i_1} j_w\}$. Thus, if $\mathcal{S}_{i_1}$ is a solution for the client i_1, then all feasible combinations of solutions for all $i \neq i_1$ are defined in a set where there exists a facility $k := k \in (\mathcal{S}_{i_1} \cup \mathcal{T}_{i_1})$.*

Proof. It is easy to see that if $\mathcal{T}_{i_1} = \emptyset$, then the only solution for all $i \neq i_1 \in \mathcal{I}$ is the set of facilities contained in $\mathcal{S}_{i_1}$. Now there exists a client $i^* \neq i_1 \in \mathcal{I}$ for which there is a facility j^* such that $j^* \notin (\mathcal{S}_{i_1} \cup \mathcal{T}_{i_1})$, then necessarily it has a ranking better than a $j \in \mathcal{S}_{i_1}$, but by definition the facilities $j \in \mathcal{S}_{i_1}$ must be the most preferred facilities for the client i_1, which is a contradiction. □

In the following, this theorem will be of vital importance for modeling the relation between the reliability of the model and the order of preference of the clients. Let x_{ijl} be a decision variable that takes value 1 if customer i is associated with the facility j at level-l of reliability, when j is the d-th preferred facility for customer i, and 0 otherwise. Let y_j, $j \in \mathcal{J}$, be 1 if a facility j is opened, and 0 otherwise.

Observe that only the preference order for each client i is of importance. We define the set $\mathcal{P}_{ij} = \{k \in J : k >_i j\}$ for all $j \in \mathcal{J}$. By Theorem 1, an optimal solution of this problem at level-1 of reliability can be found if $x_{ij1} = 1$ then $y_k = 0$, $k \in \mathcal{P}_{ij}$. For the level-2 if $x_{ij2} = 1$ then $y_k = 0$, $k \neq k_1 \in \mathcal{P}_{ij}$ and $y_{k_1} = 1$ for only one. For the level-3 of reliability, if $x_{ij3} = 1$ then $y_k = 0$ for all $k \neq \{k_1, k_2\} \in \mathcal{P}_{ij}$ and $\{y_{k_1}, y_{k_2}\} = 1$. Because for all clients $i \in \mathcal{I}$ the level-1 of assignment must have a better preference order that level-2, for level-2 of assignment must have a better preference order that level-3 and so on. Thus, we can rewrite the last result for all the client i in the equivalent form $\sum_{k \in \mathcal{P}_{ij}} y_k \leq |\mathcal{P}_{ij}|(1 - x_{ijl}) + (l - 1)$. Only a facility j with ranking $d_{ij} \in \mathcal{W}(i,l)$ can be a feasible assignation, where $\mathcal{W}(i,l) := \{j \in \mathcal{J} : d_{ij} \in [l, n - q + l]\}$. This result can be summarized in the following theorem.

Theorem 2. *Let $i \in \mathcal{I}$,$j \in \mathcal{J}$, $l \in \mathcal{L}$ and let us define the sets $\mathcal{W}(i,l) := \{j \in \mathcal{J} : d_{ij} \in [l, n - q + l]\}$, with $\mathcal{W}(i,l) \subseteq \mathcal{J}$,$\forall l$. A feasible assignation for a client i to a facility j at level of reliability l represented by x_{ijl}can only be feasible if d_{ij} belong to $\mathcal{W}(i,l)$.*

Proof. Assume that there exists other possible facility assignation k for the customer i to a l-level of reliability, then $x_{ikl} = 1$with $d_{ik} \in$

$\mathcal{J} \setminus \mathcal{W}(i,l)$. Case: $d_{ik} < d_{ij}$, it is not possible because l levels have to be assigned before. Case $d_{ik} > d_{ij}$, implies that $q - l$ levels have to be assigned, but there are only $q - d_{ik}$ free feasible facilities, which results in an infeasible solution. □

Thus, for example, for level 1 of reliability only the $j \in \mathcal{J}$, which have a ranking of preference between $[1, n - q + 1]$ can be a feasible solution, because $(n - q)$ levels have to be assigned jet, therefore, this is the minimal set. The formulation of our problem is now as follows:

$$\min \sum_{i\in\mathcal{I}} \sum_{l\in\mathcal{L}} \sum_{j:d_{ij}\in\mathcal{W}(i,l)} c_{ij} x_{ijl} \left(1 - p_{ijl}\right) \prod_{t=1}^{l-1} p_{ijt} + \sum_{j\in\mathcal{J}} f_j y_j \tag{1}$$

$$s.t \quad \sum_{j:d_{ij}\in\mathcal{W}(i,l)} x_{ijl} = 1 \qquad \forall i \in \mathcal{I}, \forall l \in \mathcal{L} \tag{2}$$

$$x_{ijl} \leq y_j \qquad \forall i \in \mathcal{I},\ \forall l \in \mathcal{L},\ j : d_{ij} \in \mathcal{W}(i,l) \tag{3}$$

$$x_{ijl} = 0 \qquad \forall i \in \mathcal{I}, \forall l \in \mathcal{L}, j : d_{ij} \in \mathcal{J} \setminus \mathcal{W}(i,l) \tag{4}$$

$$\sum_{l\in\mathcal{L}:\, d_{ij}\in\mathcal{W}(i,l)} x_{ijl} \leq 1 \qquad \forall i \in \mathcal{I},\ j \in \mathcal{J} \tag{5}$$

$$\sum_{k\in\mathcal{P}_{ij}} y_k \leq \{|\mathcal{P}_{ij}|\left(1 - x_{ijl}\right) \quad \forall i \in \mathcal{I},\ \forall l \in \mathcal{L},\ j :\ d_{ij} \in \mathcal{W}(i,l) \tag{6}$$

$$+(l-1)\}$$

$$y_j \in \{0,1\} \qquad \forall j \in \mathcal{J}$$

$$x_{ijl} \in \{0,1\} \qquad \forall i \in \mathcal{I},\ \forall l \in \mathcal{L},\ j :\ d_{ij} \in \mathcal{W}(i,l)$$

The objective function (1) is straightforward. Constraints (2) require that each customer i is assigned to a level-l facility j. Constraints (3) and (4) prohibit an assignment to a facility that has not been opened. The inequalities (5) prohibit a customer from being assigned to a given facility at more than one level. Constraints (6) model the preferences of customers. It means that, if a facility j is opened every customer i will be served by either j or by a facility which i prefers to j at level l. As in the UFLP, there is always an optimal solution in which variables x_{ijl} take binary values because the problem is uncapacitated. In this context, it does not make any sense to consider all possible scenarios, because otherwise the worst case scenario is always the one in which all facilities fail and the assignment is trivial. We might consider all scenarios in which, at most $q < n$ facilities fail, with a probability $p_{ijl} \in (0,1)$.

3 Experimental Results

The experiments were conducted as follows. For each of $n = 50$, $m \in \{50, 100\}$, $q \in \{1, 3\}$ and $p \in \{0.01, 0.1, 0.9\}$ parameters we generated random costs based in an uniform distribution with *set.seed(1)* in the statistical software R. For the costs c_{ij} on the interval $[100, 300]$ and for the fixed costs f_j on the interval $[500, 1000]$. For the client's preference we sample the rankings of the set randomly. All test have been performed on a Pentium IV , with CPU 2.4 GHz processors, 512 RAM memory under Windows XP as operative system. The solver used was Xpress optimizer version 1.17.12.

The results are reported in Table 1. *CPU T.* gives the total number of CPU seconds, *S.I* is the total number of iterations of the simplex dual algorithm, *gap %* correspond to the percentage of the difference between the optimal value of the linear relaxation and the integer problem, *#N* are the number of nodes of the branching tree and *O.f* is the optimal value of the integer problem.

Two trends are evident from this results. The first is that the complexity of the instances increases excessively when the amount of clients augment and when the number of reliability levels q is approximatly $\lfloor n/2 \rfloor$, because the number of feasible combinations of facilities increase considerably. The second trend show that in general the problems with probability to fail close to 1 are easier to solve that problems close to 0. In principle because the main parameters that contribute to the minimization of the problem are the fixed costs, when p is close to 1. However, for highly regular cost structures it is very difficult to solve it as it is well known in the literature.

Table 1. Results for the q-level UFLPP with $n = 50$, $m = \{50, 100\}$ and different values for the failure probability p and the level of reliability q

		q=1						*q=3*					
$n \times m$	p	#F	CPU T	S.I	#N	gap%	O.f	#F	CPU T	S.I	#N	gap%	O.f
	0.01	3	382	1207	499	27.18	9693.3	3	13501	6166	42149	26.07	10469.1
50×50	0.1	1	226	1247	241	11.84	8859.1	3	13154	5207	16890	17.88	10595.1
	0.9	1	1	2344	1	0	1443.9	3	11	5479	1	0.18	4178.16
		q=1						*q=3*					
$n \times m$	p	#F	CPU T	S.I	#N	gap%	O.f	#F	CPU T	S.I	#N	gap%	O.f
	0.01	1	5062	2335	3765	38.38	19310.6	3	195562	10194	152192	41.97	20144.3
50×100	0.1	1	3512	2023	2193	29.88	17633.2	14	140280	9775	50890	36.84	20133.8
	0.9	1	1	3799	1	0	2472.4	3	868	10948	7	7.96	6928.7

4 Conclusions

This paper presented an original approach to reliable models of facility location with client's preferences. This model attempts to find solutions that are both inexpensive and reliable for the locator, and attractive from the point of view of the clients. We have shown empirically some results for a general instance, but the difficulties to solve this formulation is in evidence. A proof of the hardness is still indispensable.

In an extend version of this paper we consider the case in which the probability of failure is dependent, where we define this dependence inside of subsets, which contain a determined amount of facilities, where this dependence is defined in relation to spatial characteristics of the facilities. Some upper bounds can be derived for the general formulation, which are useful to determine a good starting solution.

Acknowledgments

It is a pleasure of the first author to express his gratitude to Prof. Dr. Stefan Nickel and the people of the chair for Operation Research at the Saarland University for their warm hospitality and to acknowledge the support provided by the ALFA program of the European Union under the project SistIng (no. II-0321-FA).

References

1. L. Cánovas, S. García, M. Labbé, and A. Marín. A stregthened formulation for the simple plant location problem with order. *Operations Research Letters, to appear.*, 2006.
2. S. García, L. Cánovas, and A. Marin. New inequalities for the p-median simple plant location problem with order. *Submitted for Publication*, 2006.
3. E. Goncharov and Y. Kochetov. Probabilistic tabu search algorithm for the multi-stage uncapacitated facility location problem. *Operations Research Proceedings 2000,Springer, Berlin*, pages 65–70, 2001.
4. P. Hanjoul and P. Peeters. A facility location problem with clients' pereference orderings. *Regional Sci. Urban Economics*, 17:451 –473, 1987.
5. P. Hansen, Y. Kochetov, and N. Mladenovic. Lower bounds for the uncapacitated facility location problem with user preferences. *Les Cahiers du GERAD*, 24, 2004.
6. M. Menezes, O. Berman, and D. Krass. The median problem and reliability of facilities . *INFORMS Annual meeting, Atlanta*, 2002.
7. M. Menezes, O. Berman, and D. Krass. Minisum with imperfect information. *INFORMS Annual Meeting, Atlanta*, 2003.
8. L. Snyder and M. Daskin. Reliability models for facility location: The expected failure cost case. *Trasnportation Sience*, 39:400 – 416, 2005.

The Real-Time Vehicle Routing Problem

Irena Okhrin and Knut Richter

Department of Industrial Management, European University Viadrina
Grosse Scharrnstrasse 59, 15230 Frankfurt (Oder), Germany
irena.okhrin@euv-frankfurt-o.de, richter@euv-frankfurt-o.de

1 Introduction

Vehicle routing problems (VRPs) appear in distributing and/or collecting of goods by commercial or public transport. The aim of a VRP is to determine a route and schedule of vehicles in order to satisfy customer orders and minimize operational costs. In the past, vehicles executing routes and dispatchers in the control center were acting separately, without or with only little information exchange. The position of vehicles en route was not known to the dispatcher and it was not always possible to establish a good connection with drivers. Recent advances in information and communication technologies improve dramatically the quality of communication between drivers and the dispatching office. New customer orders as well as route changes can now be easily communicated to drivers, thus enhancing service quality and reducing costs. Moreover, state-of-the-art navigation systems provide real-time traffic and weather conditions allowing to escape hampered roads.

2 Problem Description

We consider a vehicle routing problem with online travel time information. The problem is defined on a complete graph $G = (V, A)$, where V is the vertex set and A the arc set. Vertex 0 represents a depot whilst other vertexes represent geographically dispersed customers that have to be served. A positive deterministic demand is associated with every customer. The demand of a single customer cannot be split and should be serviced by one vehicle only. Each customer defines its desired period of time when he wishes to be served. A set of K identical vehicles

with capacity Q is based at the single depot. Each vehicle may perform at most one route which starts and ends at the depot. The vehicle maximum load may not exceed the vehicle capacity Q. The objective is to design routes on G such that every customer belongs to exactly one route and the total travel time of all vehicles is minimized.

The frequently considered constant travel time function is not realistic. In practice, it fluctuates because of changing traffic and weather conditions like, for example, congestion during rush hours, accidents, etc. Furthermore, available models for the dynamic vehicle routing usually imply that a vehicle en route must first reach its current destination and only after that it may be diverted from its route. Exactly on the way to its immediate destination, however, the vehicle may encounter an unpredicted congestion or other traffic impediment. So, the vehicle has to wait unreasonably long, instead of deviating from its route and serving other customers in the meantime.

Thanks to mobile technology we can overcome the mentioned shortcomings and model vehicle routing in more realistic settings. State-of-the-art mobile technologies substantially facilitate the dynamic vehicle routing. First, they allow locating vehicles in real time. This gives the decision center the overview over the routes execution. Second, they enable the online communication between the drivers and the dispatching center. Thus, new instructions can be sent to drivers at any time, regardless of their location and status. And finally, mobile technologies are capable to capture varying traffic conditions in real time and in the short run predict with high accuracy the travel time between a pair of nodes. All these factors enable modelling approaches that even better approximate the real-world conditions.

We formulate the real-time VRP as a series of static vehicle routing problems with heterogeneous fleet at a specific point of time. We use the concept of time rolling horizon and run a re-optimization procedure to find new vehicle routes every time when the travel time between a pair of nodes is updated. For the vehicles that at time of routes adjustment are in transit to their destinations we create artificial intermediate nodes. The re-optimization algorithm is then performed on the graph that includes also the artificial intermediate nodes.

3 Solution Algorithm

The time-dependent vehicle routing problem is a generalization of the classical travelling salesman problem (TSP) and thus belongs to the class of NP-complete problems [4]. Exact solution algorithms can solve

to optimality only small instances of the problem, working unreasonably long for larger problems. Hence, to solve the described problem we implemented a genetic algorithm metaheuristics. Genetic algorithms were successfully deployed to VRPs and have proved to produce good quality solutions [1, 3, 5, 8].

Initial population. Unlike many heuristics, a genetic algorithm works with groups of solutions, instead of considering one solution at the time. Therefore, an initial population of feasible solutions has to be generated at the beginning of the algorithm performance. We develop a fast and effective method to initially assign all customers to routes. At first we sort the customers by the starting time of the time window. The customer with the earliest starting time is taken as the first customer in the first route. Further customers are chosen randomly one after the other and appended to the route until the time schedule and capacity constrains are satisfied. If after hundred attempts no valid customer for the given route can be chosen, we initiate a new route. ¿From the rest of the customers we again select the one with the earliest time window starting time and set this client as the first for the next route. The procedure is repeated until no unserved customers are left and enough individuals for the initial population are created.

Selection criteria. To select a set of parents for further reproduction we implement the stochastic tournament selection operator [2, p. 88]. The core of the operator is a tournament set which consists of k individuals randomly chosen from the population. These individuals are replaced in the population, what increases the variance of the process and favours the genetic drift.

Crossover operator. We adopt the special crossover operator called best cost route crossover (BCRC) [8], which is particularly suitable for VRP with hard time windows. The operator produces two feasible offspring from two parents p_1 and p_2 by executing the following procedure. In the first step, a random route is selected from each parent (route r_1 is selected from parent p_1 and r_2 from p_2). Then the customers that belong to the route r_2 are removed from the parent p_1. Analogously, the customers belonging to the route r_1 are removed from parent p_2. To yield the feasible children, the removed customers should be selected randomly and re-inserted back into the corresponding solution at the least cost. For that purpose the algorithm scans all possible locations for insertion and chooses the feasible ones. The removed customer is then inserted into the place that induces the minimum additional costs. If no feasible insertion place can be found, a new route containing the removed customer alone is created and added to the offspring.

Mutation operator. Finally, a mutation operator is applied to the population to ensure that the algorithm does not converge prematurely to a local optimum. As mutation introduces a random alteration to diversify the search, it can be a relatively destructive element, deteriorating the fitness of the solution. Therefore, the mutation operator is applied to only small fraction of the offspring, determined by the mutation rate. We applied a widely-used swap mutation algorithm, exchanging two customers with similar time windows [1].

Construction of a new population. In the new generation the offspring created by the sequential application of the selection, crossover, and mutation operators, completely replace the parents. Only the small number of the worst offspring are left aside and instead of them the best individuals from the old generation, called *elite*, are included into the new generation. Such strategy is called elitism [6, p. 168]. It ensures that the best solutions can propagate through generations without the effect of the crossover or mutation operators. Therefore, the fitness value of the best solution is monotonically nondecreasing from one generation to another [2, p. 91]. In the new generation, however, the elite individuals have to compete with the fittest offspring, forcing the algorithm to converge towards an optimum.

4 Computational Results

The proposed genetic algorithm was tested in two stages: Stage one with constant travel times and stage two with variable travel times. Even though the considered problem is dynamic and time-dependent, the algorithm was initially tested on the constant travel time data to prove its efficiency. For this purpose we take the Solomon's benchmark problems with the long scheduling horizon [9]. The received results for the constant travel time tests are comparable with best known so far. In fact, for eight instances out of eleven from the random problem set and for five instances out of eight from the semi-clustered problem set we were able to outperform the best known solutions. For more details about constant travel time test please see [7].

The second stage of the computational experiments simulates the vehicle routing in more realistic settings. Here we assume that travel times between a pair of nodes undergo two types of disturbances. On the one hand, a link travel time function depends on the time of day when a vehicle drives along this link. Thus we capture time dependency due to periodic traffic congestions which is based on historic data and hence known a priori. On the other hand, we incorporate unpredicted short-

Table 1. Test results for real-time travel times

Problem	Average with re-optim.	Average without re-optim.	Rejected in %	Problem	Average with re-optim.	Average without re-optim.	Rejected in %
R201	1211.06	1218.51	15	RC201	1319.02	1320.84	0
R202	1084.57	1111.92	20	RC202	1148.95	1168.27	5
R203	910.07	929.85	0	RC203	987.44	995.53	20
R204	759.15	757.26	5	RC204	817.23	829.63	10
R205	1003.64	1022.87	10	RC205	1197.12	1179.83	0
R206	910.32	915.43	15	RC206	1098.97	1130.09	15
R207	831.21	836.95	15	RC207	1021.00	1021.01	10
R208	731.18	732.54	20	RC208	825.43	834.00	5
R209	890.04	896.07	10				
R210	948.58	956.41	5				
R211	794.88	811.11	10				

term fluctuations of travel times that occur due to unexpected dynamic events like accidents. Therefore, we assume that the dispatching centre has the real-time overview over traffic conditions. Based on these data it updates the optimal solution and periodically adjusts the vehicle routes in order to avoid hampered roads.

The test results for the real-time case are presented in Table 1. Column "Average with re-optim." contains average travel time value calculated over twenty runs when routes re-optimization was undertaken after every perturbation of the travel time matrix. On the contrary, column "Average without re-optim." states the results for the case when travel times are periodically updated but the routes are not correspondingly adjusted. Consequently, the vehicles have to follow the initial routes constructed at the beginning of the planning period. The value difference between the two columns shows that even for small perturbations of travel times the periodic route adjustment leads to better results. Finally, column "Rejected" indicates the fraction of problem instances containing customers that could not be served in the case without route re-optimization. This is due to the fact that the traversed routes are definitely less-than-optimal while being determined for obsolete travel times. Hence, the vehicles arrive to the customers after the ending time of the time windows and are not able to serve them. From these experiments we can see that if solutions computed for constant travel times are deployed in real-time settings, their optimality and even feasibility are subject to substantial changes. Therefore, to be able to serve all customers and decrease costs one has to make

use of modern technologies and real-time data and promptly react to the ever-changing settings of the real world.

5 Conclusions

The paper deals with a vehicle routing problem with real-time travel times. We assume the deployment of mobile information and communication system that allows us to consider time-dependent travel times which are updated on a permanent basis. Thus we incorporate the possibility to react to traffic impediments and divert a vehicle en route from its current destination. To solve the developed problem we implement a genetic algorithm. We perform an extensive computational study in order to prove the efficiency of the proposed algorithm on well-known static benchmarks as well as to test its performance in dynamic settings. The achieved results are competitive with best published solutions and prove the efficiency of the proposed solution method.

References

1. Alvarenga GB, Mateus GR, de Tomi G (2007) A genetic and set partitioning two-phase approach for the vehicle routing problem with time windows. Computers & Operations Research 34:1561–1584
2. Dréo J, Pétrovski A, Siarry P, Taillard E (2006) Metaheuristics for Hard Optimization. Springer
3. Hanshar FT, Ombuki-Berman BM (2007) Dynamic vehicle routing using genetic algorithms. Applied Intelligence 27:89–99
4. Lenstra JK, Rinnooy Kan AHG (1981) Complexity of vehicle routing and scheduling problems. Networks 11:221–227
5. Marinakis Y, Migdalas A, Pardalos PM (2007) A new bilevel formulation for the vehicle routing problem and a solution method using a genetic algorithm. Journal of Global Optimization 38:555–580
6. Mitchell M (1996) An Introduction to Genetic Algorithms. MIT Press
7. Okhrin I, Richter K (2007) The vehicle routing problem with real-time travel times. Arbeitsberichte Mobile Internet Business 8, European University Viadrina, Frankfurt (Oder)
8. Ombuki B, Ross BJ, Hanshar F (2006) Multi-objective genetic algorithms for vehicle routing problem with time windows. Applied Intelligence 24:17–30
9. Solomon MM (1987) Algorithms for the vehicle routing and scheduling problems with time window constrains. Operations Research 35:254–265

A Decision Support System for Planning Promotion Time Slots*

Paulo A. Pereira[1], Fernando A. C. C. Fontes[1], and
Dalila B. M. M. Fontes[2]

[1] Dept. of Mathematics for Science and Technology, Universidade do Minho, 4800-058 Guimarães, Portugal. ffontes;ppereira@mct.uminho.pt
[2] Faculdade de Economia da Universidade do Porto, Rua Dr. Roberto Frias, 4200-464 Porto, Portugal. fontes@fep.up.pt

Summary. We report on the development of a Decision Support System (DSS) to plan the best assignment for the weekly promotion space of a TV station. Each product to promote has a given target audience that is best reached at specific time periods during the week. The DSS aims to maximize the total viewing for each product within its target audience while fulfilling a set of constraints defined by the user. The purpose of this paper is to describe the development and successful implementation of a heuristic-based scheduling software system that has been developed for a major Portuguese TV station.

Key words: Decision support systems, scheduling, heuristics.

1 Introduction

Decision Support Systems (DSS) are used to support decision making in almost every area of business. In this work we report on a DSS specifically developed for a Portuguese TV station. As noted by Kendall and Kendall ([7], pp. 320-329), all DSS methodology can fall into the two categories of analytic or heuristic. Analytic DSS that use optimization procedures have been used to support the scheduling of personnel, equipment and even value chain activities [8]. Unfortunately, the size and time requirements to generate even small assignment schedules using an analytic method can be prohibitive. Hence, a heuristic DSS assignment schedule is proposed.

* Research supported by SIC and FCT/POCI 2010/FEDER through Project POCTI/MAT/61842/2004.

Optimization techniques have been successfully employed in various fields, however only a few studies exist that address the optimization problems in the television and media industry. The majority of these studies focus on scheduling television programs and on modelling audience and audience behaviour rather than scheduling breaks. The literature dealing specifically with TV breaks scheduling problem is sparse and deals only with commercial breaks. Bollapragada et al. [2, 3] studied the commercial scheduling problem to generate sales plans to meet the requirement of a single advertiser. The problem was modelled as an integer program and solved sequentially for each advertiser with an objective to make the least use of premium inventory. Bollapragada [4] then studied the problem on scheduling commercials over a specific period so that the airing of the same commercials are spread as evenly as possible. Jones [6] introduced the advertising allocation problem as an example to design incompletely specified combinatorial auctions where potentially hundreds of advertisers can submit combinatorial bids for the airing of their commercials in the advertising slots. The problem was modelled as an integer program and heuristics based on constraint programming were used to find feasible solutions. Based on this work, Zhang [9] proposed a two-step hierarchical approach. First, a winner determination problem is solved to select advertisers and assign them to shows. Then, a pod assignment problem is used to schedule selected advertisers' commercials to slots within a specific show. In another recent work Mihiotis and Tsakiris [5] have studied the advertising allocation problem but from the advertising company point of view. More specifically, they solve the problem of deciding where the commercials are to be place, given the set of available places, their costs, and the number of people viewing each one of them. The choices are to be made in order to have the maximum total of people viewing subject to a budget constraint. They developed a binary mathematical programming model that, due to the enormous number of variables, is then solved heuristically.

The problem considered in this paper takes as input a list of breaks, a list of show spots, and a list of requirements for shows that the marketing department would like to satisfy. The objective is to build an assignment schedule that maximizes the total viewing for each product within its target audience satisfying several constraints and fulfilling the audience targets. We name this problem TV Self-Promotion Assignment Scheduling Problem (TSPASP).

2 Problem Description and Formulation

In a TV channel week there are several self promotion breaks, that is, slots of time in between shows that are not allocated to commercial purposes. These breaks, from now on referred to simply as breaks, are to be used to advertise shows that are still to be broadcasted and also own merchandizing. There are a number of advertising campaigns, that we call spots, from which we must choose the ones to be used in the aforementioned breaks.

The TSPASP described below essentially consists in assigning a subset of the existing spots, each of which can be used more that once, to the available breaks, subject to audience, operating and legal constraints. Each break is characterized by the broadcasting time, its duration, and its forecasted audience in each segment, called target. Each existing advertising spot is characterized by the show/product (for the sake of exposition form now on we will use show only) they advertise, by its duration, and by the nature o the advertised show (i.e a spot referring to show that shows alcohol or sex may only be broadcasted after 22:30). However, there are other issues that must be accounted for. For instance, a show must not be advertised after it has been broadcasted. Associated with each show there are requirement that must be satisfied, such as number of times a show must be advertised, how show advertisements should be spread over the week, the number of people that has seen at least one spot for the show, and the number of times that a show has been advertised to its intended audience.

Let c_{it} be the number of contacts forecasted at break i for target t and let p_{jk} and q_{ij} be binary parameters denoting whether spot j is of product k and spot j is intended for target t. Since we must allocate spots to breaks, we define binary variables x_{ij} that are to be set to 1 if at break i spot j is broadcasted and set to 0 otherwise. The mathematical programming model ($\mathcal{P}$), allows to determine the spot broadcast decisions, which are made in order to maximize total viewing for each product within its target audience, as given by equation (1), and must satisfy the constraints given in equations (2) to (9).

The first 4 constraints, equations (2) to (5), are show constraints and establish that each show must have a minimum percentages $S_{k_{min}}$ of broadcasted spots within pre-specified time intervals I_l (i.e. up to 2 hours before the show being broadcasted and in each day I_d, until it is broadcasted; has advertising maximum and minimum limits; and must have a pre-specified minimum cover $C_{kt_{min}}$ in its targeted audience. The following 2 constraints specify maximum and minimum limits for the number of times that each spot is broadcasted and also intervals of time

F_i when spot broadcasting is forbidden, either due to legal constraints or to operational ones. In equation (8) we imposed that the duration s_j of the spots broadcasted in each break does not exceed break duration b_i. Finally, the binary nature of the decisions to be made is given by equation (9).

$$
\begin{aligned}
(\mathcal{P})\ \text{Maximize} \quad & GRP's = \textstyle\sum_i \sum_j \sum_k c_{it} \cdot x_{ij} \cdot q_{jt} && (1)\\
\text{subject to} \quad & S_{k_{min}} \textstyle\sum_i \sum_j x_{ij} \cdot p_{jk} \leq \sum_{i \in I_l} \sum_j x_{ij} \cdot p_{jk} && \forall\ k, I_l. \quad (2)\\
& D_{min} \textstyle\sum_i \sum_j x_{ij} \cdot p_{jk} \leq \sum_{i \in I_d} \sum_j x_{ij} \cdot p_{jk} && \forall\ k, I_d. \quad (3)\\
& K_{min} \leq \textstyle\sum_i \sum_j x_{ij} \cdot p_{jk} \leq K_{max} && \forall\ k. \quad (4)\\
& \textstyle\sum_i \sum_j x_{ij} \cdot p_{ik} \cdot c_{it} \geq C_{kt_{min}} && \forall\ k, t\ . \quad (5)\\
& S_{min} \leq \textstyle\sum_i x_{ij} \cdot p_{jk} \leq S_{max} && \forall\ j. \quad (6)\\
& \textstyle\sum_j x_{ij} = 0 && \forall\ j \in F_i (7)\\
& \textstyle\sum_j s_j \cdot x_{ij} \leq b_i && \forall\ i. \quad (8)\\
& x_{ij} \in \{0, 1\}0 && \forall\ i, j. \quad (9)
\end{aligned}
$$

3 Methodology

The methodology proposed is a decision support system, that we have named PlanOptimUM, which includes a heuristic procedure to generate solutions. After discarding the non feasible solutions, the remainder are evaluated. The best solutions are then suggested to the operator that through editing can include some extra elements, not provided to the PlanOptimUM. These changed solutions can then be re-evaluated in order to choose the most convenient one.

Solutions Generation: In order to generate solutions we have implemented an heuristic procedure that outputs spot-break assignment binary matrices. The solution procedure has basically 4 stages: In stage (i) the show-defined constraints are converted into spot-constraints. Therefore, except for the break duration constraints which are dealt with differently, we only have spot-constraints. Although some constraints are easily converted, others have required complex procedure to do so. This is the case of the minimum coverage constraints. We have developed an estimator for the number of times each show would need to be advertised since the coverage is a nonlinear and unknown function of the broadcasted spots. In stage (ii) we compute the maximum Mb_j and the minimum mb_j number of times that spot j may be used. In stage (iii) we generate the binary matrix by resorting to an iterative greedy heuristic procedure based on the following

- select the spot having the largest value of the remaining times to be used. (Initially mb_j.)

- from the breaks that still have free time, select the break having the highest audience rate for the target of selected spot.

In stage (iv) even if the solution is feasible (which typically happens) we look for breaks with available time in order to improve the solution. For each break we iteratively select a spot/show for which the target has the highest audience rating whose duration is compatible with the remaining break time.

User Interface: The software system used in the development of the computer programs that comprise the PlanOptimUM was MATLAB and C for model solving and Visual Basic for the interface.

Given the complexity of the problem many of the objectives that management wanted to maximize and minimize have been converted into constraints. In seeking a maximum amount of autonomy in the use of this system the interface developed allows for constraint introduction. Furthermore, the PlanOptimUM, through the use of the interface can easily be used to experiment other schedules and scenarios without consuming much time.

PlanOptimUM generates the spot-break assignment schedules. The schedules can then be edited by an operator, that can directly alter in order to incorporate his/her own personal and subjective judgment into the assignment process. Editing and customization features of the system included the ability to override the schedule and make pre-assignments of spots to breaks.

In Figure 1 we show how the interface looks like, when the operator is changing show characteristics and editing the solution.

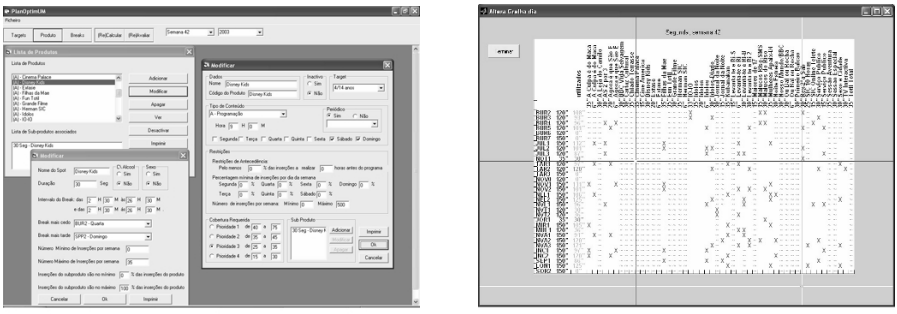


Fig. 1. Solution tuning

4 Conclusions

The application of the PlanOptimUM took place at the SIC-Sociedade Independente de Comunicação SA, in Carnaxide, Portugal. SIC is an over-the-air commercial television that frequently leads audience shares in Portugal. The management of SIC intended to use the PlanOptimUM to prepare weekly schedules for spot assignments to the breaks.

On average there are about 50 shows and products to be advertised each week and the number of different spots to advertise each one of them varies between 1 to 5. On one week there are about 230 self promotion breaks to which spots must be assigned to. SIC management wanted a reduction in the time required to establish weekly spot assignment schedules. This has been greatly achieved since the work of more than one person during a week is now done by the one person in one afternoon. A comparison of three months observations on both manually generated schedules and the PlanOptimUM generated schedules has shown a substantial qualitative improvement in scheduling.

A natural improvement, which is already being explored, is to include in this system optimization scheduling techniques.

References

1. Benoist T., Bourreau E., Rottembourg B.: *The TV-Break Packing Problem,* European Journal of Operational Research 176 (3):1371-1386, 2007.
2. Bollapragada S., Cheng H., Philips M., Garbiras M., Scholes M., Gibbs T., and Humhreville M.: *"NBC"s Optimization Systems Increase Revenues and Productivity*, Interfaces 32 (1): 47-60, 2002.
3. Bollapragada S., Garbiras M.: *Scheduling commercials on broadcast television*, Operations Research 52 (3): 337-345, 2004.
4. Bollapragada S., Bussieck M.R., Mallik S.: *Scheduling commercials videotapes in broadcast television*, Operations Research 52 (5): 679-689, 2004.
5. Mihiotis A., Tsakiris I.: *A mathematical programming study of advertising allocation problem,* Applied Mathematics and Computation 148 (2): 373-379, 2004.
6. Jones J. J., Tsakiris I.: *Incompletely Specified Combinatorial Auction: An Alternative Allocation Mechanism for Business to Business Negotiations,* Ph.D. Dissertation, University of Florida, FL, 2000.
7. Kendall K.E. and Kendall J.E.: *Systems Analysis and Design,* (Prentice Hall, Englewood Cliffs, NJ, 1988.
8. Shapiro J.F., Singhal V.M. and Wagner S.N.: *Optimizing the Value Chain,* Interfaces 23 (2): 102- 117, 1993.
9. Zang Y: *Mathematical models for the television advertising allocation problem,* International Journal of Operational Research 1, 302-322, 2006.

Greedy Heuristics and Weight-Coded EAs for Multidimensional Knapsack Problems and Multi-Unit Combinatorial Auctions

Jella Pfeiffer and Franz Rothlauf

Dept. of IS; University Mainz
jella.pfeiffer@uni-mainz.de, rothlauf@uni-mainz.de

1 Introduction

The multidimensional knapsack problem (MDKP) is a generalized variant of the $\mathcal{NP}$-complete knapsack problem (KP). The MDKP assumes one knapsack being packed with a number of items x_j so that the total profit $\sum p_j$ of the selected items is maximized. In contrast to the standard KP, each item has m different properties (dimensions) r_{ij} $(i = 1, \ldots, m; j = 1, \ldots, n)$ consuming c_i of the knapsack:

$$maximize \ \sum_{j=1}^{n} p_j x_j \tag{1}$$

$$subject \ to \ \sum_{j=1}^{n} r_{ij} x_j \leq c_i, \ i = 1, ..., m \tag{2}$$

$$with \ \ x_j \in \{0, 1\}, \ j = 1, ..., n, \ p_j, c_i \in \mathbb{N}, \ r_{ij} \in \mathbb{N}_0.$$

A number of relevant real-world problems can be modelled as MDKPs such as allocation problems, logistics problems, or cutting stock problems [6]. Recently [4], it has been noticed that also the winner determination problem (WDP) in the context of multi-unit combinatorial auctions (MUCA) can be modelled as MDKP. MUCAs are combinatorial auctions (CA) where multiple copies of each good are available. In CAs, bidding is allowed on bundles of goods, which allows bidders to express synergies between those goods they want to obtain. First, the agents submit their bids and then, the auctioneer allocates the goods to the agents so that his revenue is maximized. The revenue is the sum of all submitted bids which are accepted by the auctioneer. This alloca-

tion problem is called the WDP[1]. The application of CAs is of special interest for logistics, procurement, scheduling problems, and others.

As the literature had ignored the close relationship between MDKP and WDP for a long time, two independent lines of research have emerged. In WDP literature, mostly small test instances are used which can be exactly solved, whereas MDKP literature considers more complex instances, that are mainly solved with heuristic optimization. Since complex bidding structures have to be considered for MUCAs, this paper compares and studies the heuristic optimization approaches used in the two different research communities for the more difficult MDKP test instances. We focus on different greedy heuristics and Raidl's weight-coded evolutionary algorithm (EA) [7], and examine the EA's trade-off between the quality of the solution and the number of generations.

2 Heuristic Optimization Approaches

2.1 Literature Review

For the MDKP, heuristic optimization methods such as tabu search, ant colony optimization, or EAs have been applied. The two current state-of-the-art EAs are one with repair steps using a direct encoding (proposed by Chu and Beasley [1]), and a weight-coded approach (proposed by Raidl [7]). Other promising approaches are hybrids such as the combination of EAs with tabu search[11], linear programming [12], or branch and bound [2].

While many heuristic optimization methods have been applied for the MDKP, research on CA has focussed on exact algorithms. In heuristic optimization, some simple greedy heuristics have been discussed [4, 13], stochastic local search [5], limited discrepancy search [10], and a simulated annealing approach [3]. However, these approaches are very limited and only [4] considered the multi-unit problem extension.

Since this paper compares heuristics from both MUCA and MDKP literature, we employ greedy heuristics for CAs that can easily be extended to MUCAs and one of the MDKP state-of-the-art methods, namely the EA from Raidl [7]. Both methods can also be combined.

2.2 Greedy Heuristics

A common heuristic is the so-called *primal greedy heuristic*, which first sorts all items (bids) in decreasing order according to some criteria

[1] The profit p_j used in the MDKP corresponds to the price of bid j, while the resource consumption r_{ij} corresponds to the number of units of good i requested in bid j. The decision variable x_j denotes whether bid j wins (is accepted by the auctioneer) or looses (is not accepted by the auctioneer).

(usually r_{ij}, p_j, or a combination of both) and then adds the items one by one to the knapsack as long as no restrictions are violated. There are four common ways for sorting the items. All are easy to implement and fast.

- **Normalized Bid Price (NBP)**: $NBP_j := \frac{p_j}{(\sum_{i=1}^{m} r_{ij})^l}, \quad l \geq 0.$
- **Relaxed LP Solution (RLPS):** This approach calculates the optimal solution of the relaxed problem ($x_j \in [0,1]$) and sorts the continuous decision variables (x_j) in a decreasing order.
- **Scaled NBP (SNBP)**: *Relevance values* μ_i measure the scarcity of capacities. The underlying idea is to choose a high μ_i for dimensions with low c_i. $\mu_i = 1/c_i$ [6] results in: $SNBP_j := \frac{p_j}{\sum_{i=1}^{m} \frac{r_{ij}}{c_i}}$
- **Shadow Surplus (SS)**: *Surrogate multipliers* a_i are taken as relevance values μ_i by aggregating all capacity constraints to a single one. A straightforward approach consists in using the dual variables of the relaxed LP which serve as an approximation of how valuable a unit of a good is (and are therefore called shadow prices). We define the shadow surplus of a bid j as $SS_j := \frac{p_j}{\sum_{i=1}^{m} a_i r_{ij}}$, where a_i ($i \in \{1, \ldots, m\}$) are the solutions of the dual LP.

2.3 Weighted Encodings

In the EA proposed by [7], the genotypes are real-valued vectors $(w_1, \ldots, w_n)$, where the weight at position j indicates the importance of the jth item. A decoder constructs a solution from the genotype in two steps. In the first step, the profits p_j of the items are biased according to the corresponding w_j either as $p'_j = p_j w_j$ or $p'_j = p_j + w_j$. Subsequently, the items are ordered in decreasing order according to p'_j. In the second step, a primal greedy heuristic (see Sect. 2.2) is applied to the sorted list to obtain a complete problem solution.

Based on extensive experiments, Raidl chose the SS heuristic and biased weights of the form $p'_j = p_j w_j = p_j(1+\gamma)^{\mathcal{N}(0,1)}$ during the initialization and mutating steps. This favors small weights due to the use of a normal distribution $\mathcal{N}$. Raidl recommended setting γ to 0.05. Standard uniform crossover is applied with the probability 1 and mutation with the probability $3/n$. A steady-state EA with a population of 100 is used with phenotypic duplicate elimination. EA runs are stopped after evaluating 100,000 individuals without finding a new best solution.

The indirect representation used by Raidl results in a strong bias toward solutions which are constructed using the underlying greedy heuristic. This bias might mislead the search if the optimal solution is

not very similar to the solution generated by applying just the greedy heuristic [9]. Additional problems can occur due to the redundancy of the encoding, as many mutations of w_j do not result in a new phenotype. This problem can be addressed by using duplicate elimination (as proposed by Raidl) resulting in an additional search effort.

3 Experiments

We compare the solution quality of Raidl's EA to the heuristics from Sect. 2.2. Furthermore, the influence of structural properties of the test instances (the tightness ratio) on the algorithm's performance is studied. Finally, the trade-off between the solution quality and the running time of the EA is analyzed and inefficiencies are revealed.

All experiments were run on a Linux machine (2.2 GHz and 2GB RAM). We used the 270 MDKP test instances from the OR library[2]. They consist of 30 instances for each of the nine combinations of $m = \{5, 10, 30\}$ and $n = \{100, 250, 500\}$, from which only the ones with $m <= 5$ or $n <= 100$ can be solved optimally with CPLEX 9.0. The structure of instances is described by the *tightness ratio* $\alpha_i = \sum_{j=1}^{n} \frac{c_i}{r_{ij}}$, which expresses the scarcity of capacities. A tightness ratio of 0.25, for instance, expresses that only about 25% of the bids can be satisfied (WDP), or about 25% of all available goods can be packed into the knapsack (MDKP). Lower tightness ratios indicate more restricted problem instances.

Method performance is measured by the gap between the best found solution and the optimal solution (gap), if known, and the relaxed optimal solution ($gap', x_j \in [0, 1]$). Table 1 summarizes some of the experimental results. We only show running times for the EA, since running times of heuristics are negligible.

The results reveal the good performance of the primal greedy heuristic despite their simplicity. RLPS ($gap = 0.93$ %) and SS ($gap = 1.56$ %) are the best heuristics. As expected [7], the gaps are lower for the weight-coded EA than for the heuristics. However, the EA need a running time of up to 364s for complex instances ($m = 30,\ n = 500$).

Table 1 indicates that the gaps decrease with larger α. This influence was also partially studied by [8] who only considered gap', and concluded that there is no general evidence that the performance of greedy heuristics increases with larger α. Instead, they argued, the trend is due to the tighter bound of the relaxed LP for higher α. In contrast to these results, we are able to confirm with an ANOVA for

[2] `http://people.brunel.ac.uk/~mastjjb/jeb/info.html`

Table 1. Performance of greedy heuristics and Raidl's EA. Each row represents the average over 10 instances (due to place restrictions not all results can be printed). In the last row, the average over all 270 instances is displayed

			NBP[%]		SNBP[%]		RLPS[%]		SS[%]		Raidl[%]		
m	n	α	gap'	gap	gap'	gap	gap'	gap	gap'	gap	gap'	gap	time[s]
5	100	.25	9.611	8.706	7.453	6.525	2.452	1.478	2.530	1.557	1.011	0.023	33.26
		.50	4.595	4.163	3.203	2.764	1.283	0.835	1.621	1.175	0.465	0.014	36.71
		.75	2.696	2.386	1.695	1.381	0.877	0.561	0.981	0.665	0.325	0.007	35.98
10	250	.25	7.912	-	6.005	-	1.429	-	1.998	-	0.653	-	132.87
		.50	3.952	-	3.435	-	0.737	-	0.924	-	0.294	-	125.33
		.75	1.852	-	1.575	-	0.383	-	0.516	-	0.169	-	102.55
30	500	.25	6.972	-	5.515	-	1.259	-	2.155	-	0.884	-	356.4
		.50	3.199	-	2.829	-	0.659	-	0.979	-	0.356	-	294.13
		.75	1.624	-	1.542	-	0.315	-	0.603	-	0.206	-	364.74
average			5.039	4.974	3.897	3.592	1.272	0.931	1.864	1.560	0.617	0.06	154.37

a level of significance of 5% that with increasing α, the *gap* decreases significantly ($M = 3.74$ vs. 1.86 vs. 1.07, $F = 8.749$, $p < 0.001$, n=150). As a dependent value we used *gap* averaged over all five algorithms.

Finally, we analyze the trade-off between solution quality and running time. We postulate that excellent solutions are already created during initialization, while EA performance in later generations is low.

Figure 1 shows the average gaps over the number of generations for three MKNAPCB test instances. The results indicate that a population size of $N = 100$ already results in very small avg. gaps (1.59%) in the initial population. Initial solutions have high fitness as they are only slightly different from the solutions generated by the underlying greedy heuristic. In the first few generations, the gap decreases and stays approximately constant after a few hundred or thousand generations (fitness evaluations). Therefore, the high number of fitness evaluations proposed by Raidl [7] is not necessary and can be greatly reduced with only a minor effect on the resulting solution quality.

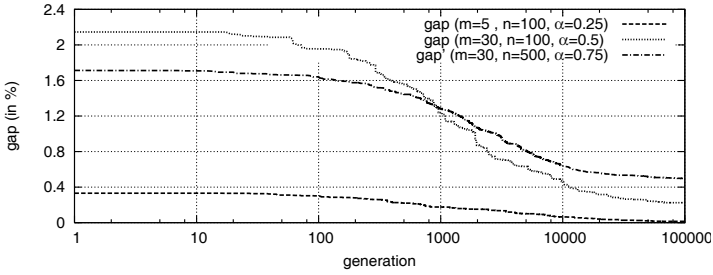


Fig. 1. Raidl's EA: Solution quality vs. running time

4 Conclusions

Since simple and fast greedy heuristics show an excellent performance, we recommend using simple local search nearby the solution of a greedy heuristic instead of putting additional effort in more complex and time-consuming algorithms, like EAs. Furthermore, since the main quality improvement of EAs occur at early stages of a run, the high running times proposed by [7] can be greatly reduced with only minor effects on the resulting solution quality.

References

1. P. C. Chu and J. E. Beasley. A genetic algorithm for the multidimensional knapsack problem. *Journal of Heuristics*, 4(1):63–86, 1998.
2. J. E. Gallardo, C. Cotta, and A. J. Fernández. Solving the multidimensional knapsack problem using an evolutionary algorithm hybridized with branch and bound. In *Proc. of IWINAC-2005*:21–30, 2005.
3. Y. Guo, A. Lim, B. Rodrigues, and Y. Zhu. A non-exact approach and experiment studies on the combinatorial auction problem. In *Proc. of HICSS-38*:82–89, 2005.
4. R. C. Holte. Combinatorial auctions, knapsack problems, and hill-climbing search. In E. Stroulia and S. Matwin, editors, *Canadian Conference on AI*:57–66. Springer, 2001.
5. H. H. Hoos and C. Boutilier. Solving combinatorial auctions using stochastic local search. In *Proc. of AAAI-00*:22–29. AAAI, 2000.
6. H. Kellerer, U. Pferschy, and D. Pisinger. *Knapsack Problems*. Springer, 2004.
7. G. Raidl. Weight-codings in a genetic algorithm for the multiconstraint knapsack problem. In *Proc. of CEC99*:596–603. IEEE Press, 1999.
8. G. Raidl and J. Gottlieb. Empirical analysis of locality, heritability and heuristic bias in evolutionary algorithms: A case study for the multidimensional knapsack problem. *ECJ*, 13(4):441–475, 2005.
9. F. Rothlauf and D. E. Goldberg. Redundant representations in evolutionary computation. *Evolutionary Computation*, 11(4):381–415, 2003.
10. Y. Sakurai, M. Yokoo, and K. Kamei. An efficient approximate algorithm for winner determination in combinatorial auctions. In *Proc. of EC-00*:30–37, 2000.
11. J. Thiel and S. Voss. Some experiences on solving multiconstraint zero-one knapsack problems with genetic algorithms. *INFOR*, 32(4):226–242, 1994.
12. M. Vasquez and J.-K. Hao. A hybrid approach for the multidimensional 0-1 knapsack problem. In *Proc. of IJCAI-01*:328–333, 2001.
13. E. Zurel and N. Nisan. An efficient approximate allocation algorithm for combinatorial auctions. In *Proc. of EC-01*:125–136. ACM, 2001.

A Metaheuristic for the Periodic Location-Routing Problem

Caroline Prodhon

Institut Charles Delaunay - FRE CNRS 2848, Université de Technologie de Troyes BP 2060, 10010 Troyes Cedex, France. caroline.prodhon@utt.fr

Summary. The well-known Vehicle Routing Problem (VRP) has been generalized toward tactical or strategic decision levels of companies but not both. The tactical extension or Periodic VRP (PVRP) plans trips over a multi-period horizon, subject to frequency constraints. The strategic extension or Location-Routing Problem (LRP) tackles location and routing decisions simultaneously as in most distribution systems interdependence between these decisions leads to low-quality solutions if depots are located first, regardless the future routes. Our goal is to combine for the first time the PVRP and LRP into the Periodic LRP or PLRP. A metaheuristic is proposed to solve large size instances of the PLRP. It is based on our Randomized Extended Clarke and Wright Algorithm (RECWA) for the LRP and it tries to take into consideration several decision levels when making a choice during the construction of a solution. The method is evaluated on three sets of instances and results are promising. Solutions are compared to the literature on particular cases such as one-day horizon (LRP) or one available depot (PVRP).

Key words: Heuristic, Periodic Location-Routing Problem.

1 Introduction

Researches have shown that separation of location and routing decisions often leads to suboptimal solutions [11]. The Location-Routing Problem (LRP) overcomes this drawback. However, only very recent papers consider both capacitated routes and depots [13, 2, 8, 7, 9].

Beside the strategic aspect of depot location, a focus on tactical decision such as Vehicle Routing Problems (VRP) leads to consider some extensions. One of these consists in integrating frequency constraints on visited customers under a given multiperiod horizon. The resulting problem is known as periodic VRP or PVRP. The methods used to solve PVRP are mainly heuristics [4, 12, 3, 6].

In this paper, the LRP and the PVRP are combined for the first time into an even more realistic problem: the periodic LRP or PLRP. It is defined on an horizon H and a complete, weighted and undirected network $G = (V, E, C)$. V is a set of nodes comprised of a subset I of m possible depot locations and a subset $J = V \backslash I$ of n customers. c_{ij} is the traveling cost between any two nodes i and j. A capacity W_i and an opening cost O_i are associated with each depot site $i \in I$. Each customer $j \in J$ has to be served with a given frequency over H, and $Comb_j$ is its set of possible combinaisons of serviced days. d_{jlr} is the demand of customer j on the day l of combinaison $r \in Comb_j$. A set K of identical vehicles of capacity Q is available each day. A vehicle used at least once from a depot during H incurs a fixed cost F and it can perform one single route per day. The following constraints must hold: i) each customer j must be served exclusively on each day l of exactly one of the combinaison $r \in Comb_j$ by one vehicle and at the amount d_{jlr}; ii) the number of routes performed by day must not exceed N; iii) each route must begin and end at the same depot within the same day and its total load must not exceed Q; iv) the total load of the routes assigned to a depot on any day $l \in H$ must fit the capacity of that depot. The objective is to find which subset of depots to open, which combinaison of visit days to assign to each customers and which routes to perform, in order to minimize the total cost. The PLRP is *NP*-hard since it reduces to the VRP when $m = 1$ and $|H| = 1$.

The proposed method to solve it is a metaheuristic that tries to keep a global vision on the problem by taking into consideration several decision levels when making a choice during the construction of a solution. Section 2 describes the framework of the proposed algorithm. The performances of the method are evaluated in Section 3. Some concluding remarks close the paper.

2 Iterative Metaheuristic

2.1 Depot Location

The location phase of the algorithm begins by considering a fictive day during which the whole set of customers has to be served. The proposed aggregation of the data comes from the fact that the same depots has to be opened all over H. Thus, we should choose the depots in accordance to the entire set J. A fictive capacity, equal to $W_i \times |H|$, is given to each depot $i \in I$. The demand of customer j is taken as its total demand over H divided by its required visit frequency.

The resulting problem on that fictive day is an LRP solved using RECWA. This is a generalization of the classical Clarke and Wright algorithm [5] for the multidepot case in which a set SD of depots is available. RECWA keeps a global view on the data during the construction of a solution. It is applied in its diversification mode with the same parametrization as proposed in [8] except that at the beginning of each construction of the bunch of flowers, $|SD| = 1$ (instead of $|SD| = 2$). $NbDivMax$ calls to RECWA are executed and the depots appearing in the best solution then obtained are kept opened during the end of the global iteration of the metaheuristic.

2.2 Combination Allocation

Once a subset of depots is opened, the aim is to construct a feasible solution for the PLRP by assigning the customers to a visit combinaison while taking into account the routing part. We assume that consecutive customers in a solution of the LRP, solved on the fictive day during the location phase, have great chance to be successive in the PLRP, whether one of their combinaisons of visit days allows it. Thus, edges linking customers, that appear in solutions of the LRP from the location phase, are recorded in a list L sorted in decreasing frequency order. The construction of a solution for the PLRP operates by trying to iteratively insert the first elements of L (with a given probability $prob$ to avoid premature convergence, and beginning by the most frequently used edges). Most of the time, when combinaisons are not already assigned to both customers involved by the edge, several possibilities have to be evaluated. If feasibility holds, the chosen assignment is the one inserting the edge in a maximum number of days at minimum cost.

The procedure stops when all the customers have an assigned visit combinaison or when the whole list L has been explored. If some customers are not in the solution, new dedicated routes are opened to serve them in such a way that the total cost is minimized. A feasible solution is then available. It is improved by a local search LS dedicated to the LRP used in [8, 9, 7], applied on each day $l \in H$.

2.3 Routing

At this point, we have a multidepot VRP by day. An intensification on the routing intends to improve the results from the day combination assignment. The RECWA is thus run on its intensification mode [8] on each period of H, followed by LS. The solution is recorded if it improves the best one found until now. This scheme iterates $NbIntMax$ times.

2.4 Local Searches for the PLRP

In addition to LS applied on each day, we propose two local searches having a view over the horizon. The first one intends to find a new combinaison of visit days to customers that reduces the routing cost. The move is performed if the best insertion cost of the customers in the new combinaison is lower than the cost to serve it in the current one. Of course, the capacity constraints must hold to accept the move.

The second technique tries to reduce the number of vehicles assigned to a depot over H. Let t_{il} be the current number of routes beginning from depot i on day l and $T_i = \max_{l \in H} t_{il}$. The aim is, for each $i \in I$, to try to reduce T_i without increasing $T_j, \quad \forall j \in I \backslash \{i\}$. Let R_{il} be the set of routes from depot i on day l, $r \in R_{il}$, and p and q be respectively the first and last customers in r. Thus, on each day l such as $t_{il} = T_i$, we evaluate for each $j \in I \backslash \{i\}$: $g_{jl} = c_{jp} + c_{qj} - c_{ip} - c_{qi}$ providing that the capacity constraints are respected and the current $t_{jl} < T_i - 1$. If it is possible to evaluate such a move on every day l with $t_{il} = T_i$, then the possible saving Δ is calculated as shown below. The routes are reassigned if Δ is positive, and LS is applied.

$$\Delta = F - \sum_{l \in H | t_{il} = T_i} G_l \quad with \quad G_l = \min_{j \in I \backslash \{i\}} g_{jl}$$

3 Computational Study

3.1 Instances

The proposed method is evaluated on three sets of instances. *The first set* contains 30 LRP instances with capacitated routes and depots, that may be found at [10]. *The second set* of 30 PLRP instances have been especially generated for this study. *The third set* has 28 instances for the PVRP available at [1].

3.2 Implementation, Parameters and Algorithms Compared

The proposed algorithm is coded in Visual C++ and has been tested on a Dell PC Optiplex GX260, with a 2.4 GHz Pentium 4, 512 MB of RAM and Windows XP. The following parameters have been selected after a preliminary testing phase, to provide the best average solution values: $NbDivMax = 7$, $NbIntMax = 7$, $prob = 0.8$ and the maximal number of global iterations $NbItMax = m * |H| * 2$.

3.3 Discussion of Results

In Table 1, times T are given in seconds. The Gap is the deviation in percentage between each method and the best-known results taken as reference. On PLRP instances, these best-known results have been obtained by trebling the number of iterations. On PVRP and LRP instances, they come from the respective websites [1, 10].

The results show that the proposed method deals well with the depot location as the gaps are small on both LRP and PLRP instances. The results even improves the ones from [8] (0.73% better on average) with similar CPU times. Only 2 gaps (10.88% and 17.61%) are far from the best known solutions on 2 hard LRP instances with $n = 100$ and $m = 5$, and only one gap is high (37.02%) on a PLRP instance with $n = 50$ and $m = 5$. This confirm the importance on the choice of depots.

Gaps are often higher on PVRP instances showing how the periodic aspect is hard to deal with. However, the results are good with an average at less than 5% on instances until more than 100 customers and less than 5 periods. Note also that the proposed method is not especially designed for PVRP problems but for much more combinatorial ones.

CPU times mainly depend on $|V|$ but remain reasonable while taking decisions from several levels. Indeed, below $n = 200$, the proposed method requires less that 2 minutes, on any kind of instances.

Table 1. Results with gap to the best-known results

LRP instances			PLRP instances			PVRP instances		
n/m	T	Gap	$n/m/\|H\|$	T	Gap	$n/\|H\|$	T	Gap
20/5	0.19	0.12	20/5/5	0.63	2.04	$\leq 50/\leq 5$	1.02	5.62
50/5	2.52	1.21	50/5/5	6.55	4.99	51-100/≤ 5	6.35	4.37
100/5	19.02	1.30	100/5/5	52.45	0.44	51-100/5-8	3.41	5.51
100/10	46.20	7.22	100/10/5	117.72	1.41	$> 100/\leq 5$	44.16	4.53
200/10	331.79	2.61	200/10/5	950.66	0.44	$> 100/6$	28.40	9.15
Average	80.10	**2.56**		226.00	**2.06**		16.67	**5.84**

4 Conclusion

In this paper, we propose a metaheuristic to deal for the first time with the Periodic Location-Routing Problem (PLRP) with both capacitated depots and vehicles. The method handles several decision levels when making a choice during the construction of a solution. it is tested on three sets of instances with up to 200 customers. The results show that the proposed algorithm is able to find good solutions even on particular cases such as LRP or PVRP.

Acknowledgments

This research is partially supported by the Conseil Regional de Champagne-Ardenne (Champagne-Ardenne district council) and by the European Social Fund.

References

1. http://neo.lcc.uma.es/radi-aeb/WebVRP/, 2007.
2. S. Barreto and C. Ferreira and J. Paixão and B. Sousa Santos. Using clustering analysis in a capacitated location-routing problem. *European Journal of Operational Research*, 179:968–977, 2007.
3. M. Chao, B.L. Golden, and E. Wasil. An improved heuristic for the periodic vehicle routing pproblem. *Networks*, 26:25–44, 1995.
4. N. Christofides and J.E. Beasley. The period routing problem. *Networks*, 14:237–256, 1984.
5. G. Clarke and J.W. Wright. Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, 12:568–581, 1964.
6. J.F. Cordeau, M. Gendreau, and G. Laporte. A tabu search heuristic for periodic and multi-depot vehicle routing problems. *Networks*, 30:105–119, 1997.
7. C. Prins, C. Prodhon, and R. Wolfler-Calvo. A memetic algorithm with population management ($MA \mid PM$) for the capacitated location-routing problem. In J.Gottlieb and G. R. Raidl, editors, *Lecture Notes in Computer Science*, volume 3906, pages 183–194. Proceedings of EvoCOP2006 (Evolutionary Computation in Combinatorial Optimization: 6th European Conference, Budapest, Hungary, April 10-12, 2006), Springer, 2006.
8. C. Prins, C. Prodhon, and R. Wolfler-Calvo. Solving the capacitated location-routing problem by a GRASP complemented by a learning process and a path relinking. *4OR - A Quarterly Journal of Operations Research*, 4:221–238, 2006.
9. C. Prins, C. Prodhon, A. Ruiz, P. Soriano, and R. Wolfler Calvo. Solving the capacitated location-routing problem by a cooperative lagrangean relaxation-granular tabu search heuristic. *Transportation Science*, 2007.
10. C. Prodhon. http://prodhonc.free.fr/homepage, 2007.
11. S. Salhi and G. K. Rand. The effect of ignoring routes when locating depots. *European Journal of Operational Research*, 39:150–156, 1989.
12. C.C.R. Tan and J.E. Beasley. A heuristic algorithm for the periodic vehicle routing problem. *Omega International Journal of Management Science*, 12(5):497–504, 1984.
13. T.H. Wu, C. Low, and J.W Bai. Heuristic solutions to multi-depot location-routing problems. *Computers and Operations Research*, 29:1393–1415, 2002.

A New Formulation of the Capacitated Discrete Ordered Median Problems with $\{0, 1\}$-Assignment

Justo Puerto

Dep. Estadística e I.O. Fac. Matemáticas. Universidad de Sevilla. Spain
puerto@us.es

Summary. The capacitated discrete ordered median location model with binary assignment admit several formulations, some of them based on the binarization of the continuous models introduced in [2]. In this paper we consider a new formulation for the binary assignment problem based on a coverage approach. We derive some basic properties of the model and compare its performance with respect to previously known formulations.

1 Introduction

In the last years, the family of discrete ordered median location problems has been introduced. (See e.g. [1, 4] and [5].) Recently, the uncapacitated models, mentioned in the above references, were extended to deal with capacities in [2]. However, although the approaches in that initial paper leads to satisfactory results concerning motivations, applications and interpretations the solution times of larger problem instances were somehow poor.

The goal of this paper is to develop a new formulation, which makes use of the coverage ideas in [3], for the capacitated version of the Discrete Ordered Median Problem with binary assignment and to compare its performance versus the known formulations for the same problem in [2].

The rest of the paper is organized as follows. First we introduce the problem and give the new formulation. In Section 2.1 we recall the formulations given in [2]. Then, Section 3 is devoted to test the efficiency of the different approaches by providing extensive numerical experiments.

2 The New Formulation and First Properties

In this section, we introduce the new formulation that was first presented at *CORAL* meeting in [7].

Let A denote the given set of M sites and identify these with the integers $1, \ldots, M$, i.e., $A = \{1, \ldots, M\}$. We assume that the set of candidate sites for new facilities is identical to the set of clients. Let $C = (c_{ij})_{i,j=1,\ldots,M}$ be the given non-negative $M \times M$ cost matrix, where c_{ij} denotes the cost of satisfying the demand of client i from a facility located at site j. Let $N \leq M$ be the number of facilities to be located. Each client i has a demand a_i that must be served and each server j has an upper bound b_j on the capacity that it can fulfill. We assume further that the demand of each client must be served by a unique server.

A solution to the location problem is given by a set of N sites; we use $X \subseteq A$ with $|X| = N$ to denote a solution. Then, the problem consists of finding the set of sites X with $|X| = N$, which can supply the overall demand at a minimum cost with respect to the ordered median objective function. (See [5], [1], [3] or [2], for details.)

We first define G as the number of different non–zero elements of the cost matrix C. Hence, we can order the different values of C in non–decreasing sequence: $c_{(0)} := 0 < c_{(1)} < c_{(2)} < \cdots < c_{(G)} := \max\limits_{1 \leq i,j \leq M} \{c_{ij}\}$.

Given a feasible solution, we can use this ordering to perform the sorting process of the allocation costs. This can be done by the following variables $t_{jk} := 1$ if the $M - j + 1$–th smallest allocation cost is at least $c_{(k)}$, and 0 otherwise ($j = 1, \ldots, M$ and $l = 1, \ldots, G$).

With respect to this definition the $M-j+1$–th smallest cost element is equal to $c_{(k)}$ if and only if $t_{jk} = 1$ and $t_{j,k+1} = 0$. Therefore, we can reformulate the objective function of the CDOMP (i.e. the ordered median function), using the variables t_{jk}, as $\sum\limits_{j=1}^{M} \sum\limits_{k=1}^{G} \lambda_{M-j+1} \cdot (c_{(k)} - c_{(k-1)}) \cdot t_{jk}$.

We need to impose some sorting constraints on the t_{jk}–variables: $t_{j+1k} \leq t_{jk} \qquad j = 1, \ldots, M-1;\ k = 1, \ldots, G$. Nonetheless, we need to guarantee that exactly N plants will be opened among the M possibilities. This can be ensured by the variables $y_j := 1$ if the server at j is open, and 0 otherwise; $\forall j = 1, \ldots, M$.

Then, we ensure that demand is covered and capacity is satisfied. Thus, we introduce the variables $x_{ij} := 1$ if the client i is allocated

to plant j, and 0 otherwise; and the corresponding constraints $x_{ij} \leq y_j \quad \forall i = 1, \ldots, M,\ j = 1, \ldots, M,\ \sum_{j=1}^{M} y_j = N.$

We assume that allocation is binary. Thus, $\sum_{j=1}^{M} x_{ij} = 1, \quad i = 1, \ldots, M$. All the demands and capacities must be satisfied: $\sum_{i=1}^{M} a_i x_{ij} \leq b_j y_j, \quad j = 1, \ldots, M.$

Next, the relationship that links the variables t and x is: $\sum_{j=1}^{M} t_{jk} = \sum_{i=1}^{M} \sum_{\substack{j=1 \\ c_{ij} \geq c_{(k)}}}^{M} x_{ij}$. They mean that the number of allocations with a cost at least $c_{(k)}$ must be equal to the number of plants that support demand from facilites at a cost greater than or equal to $c_{(k)}$.

Summing up all these constraints and the objective function, the CDOMP can be formulated as

$$\text{Min} \sum_{j=1}^{M} \sum_{k=1}^{G} \lambda_{M-j+1} \cdot (c_{(k)} - c_{(k-1)}) \cdot t_{jk} \tag{1}$$

$$\text{s.t.} \quad \sum_{j=1}^{M} x_{ij} = 1, \qquad i = 1, \ldots, M \tag{2}$$

$$\sum_{i=1}^{M} a_i x_{ij} \leq b_j y_j, \qquad j = 1, \ldots, M, \tag{3}$$

$$x_{ij} \leq y_j \qquad \forall i, j \tag{4}$$

$$\sum_{j=1}^{M} y_j = N \tag{5}$$

$$\sum_{j=1}^{M} t_{jk} = \sum_{i=1}^{M} \sum_{\substack{j=1\ldots,M \\ c_{ij} \geq c_{(k)}}} x_{ij} \qquad l = 1, \ldots, G \tag{6}$$

$$t_{j+1k} \leq t_{jk} \qquad j = 1, \ldots, M-1;\ k = 1, \ldots, G \tag{7}$$

$$t_{jk} \in \{0, 1\} \qquad j = 1, \ldots, M;\ k = 1, \ldots, G \tag{8}$$

$$x_{ij} \in \{0, 1\} \qquad i = 1, \ldots, M;\ j = 1, \ldots, M \tag{9}$$

$$y_j \in \{0, 1\} \qquad j = 1, \ldots, M\ . \tag{10}$$

Since the proposed formulation contains $O(M^2)$ binary variables and $O(M^2)$ constraints, fast solution times for larger problem instances, using standard software–tools, are very unlikely .

First of all, $(CDOMP)$ admits a formulation with $y_j \in [0, 1]$ and for each optimal solution of the relaxed problem one can obtain an optimal solution of the original problem.

The above formulation admits some valid inequalities that, at times, reinforce the linear relaxation improving the lower bound and reducing the computation time to solve the problem.

The first ones are the natural inequalities $t_{jk} \geq t_{jk+1}, \quad j = 1, \ldots, M, \; k = 1, \ldots, G-1$. They come from the fact that the rows of the t-matrix are sorted. We have observed in our experiments that these constraints are not always satisfied by the optimal solution of the linear relaxation. This family of inequalities were introduced in [3].

Our next set of inequalities state that columns of the x-matrix contain at most as many ones as their corresponding columns in the t-matrix. Then, if there are r ones in any places of a x-column, since the columns in the t-matrix are ordered in non-decreasing sequence, we get the following: $\sum_{i \in S} \sum_{\substack{k=1 \\ c_{ik} \geq c_{(j)}}}^{M} x_{ik} \leq \sum_{i=1}^{r} t_{ij}, \quad \forall \; S \subseteq \{1, \ldots, M\}, \; |S| = r, \; r = 1, \ldots, M$. Note that there are an exponential number of inequalities in this family.

Another set of valid inequalities are those stating that either client i is allocated at a cost at least $c_{(k)}$ or there must exist an open plant j such that the allocation cost of client i is smaller than $c_{(k)}$. This results in: $\sum_{\substack{j=1 \\ c_{ij} \geq c_{(k)}}}^{M} x_{ij} + \sum_{\substack{j=1 \\ c_{ij} < c_{(k)}}}^{M} y_j \geq 1, \quad i = 1, \ldots, M.$

2.1 Alternative Formulations

One can adapt to the binary assignment case two of the formulations provided in [2]. The interested reader is referred to [2] for further details.

The first formulation is based on 3-indexes variables and therefore we will refer to it by 3-*indexes*. The last formulation is based on the approach by [6] and it is only valid for those cases where the λ-vector is in non-decreasing sequence. We will refer to it by *O-T*.

3 Computational Results

In order to test the performance of the three considered formulations, we propose an experimental design that consists of the following factors: 1) *Size of the problem:* M. We consider three different levels of $M = 10, \; 20, \; 30$. 2) *Number of suppliers:* N with two levels for each choice of M: $N = \lfloor M/5 \rfloor + 1, \; \lfloor M/2 \rfloor$. 3) *Type of problem:* Each λ-vector is associated with a different objective function. Its levels are designed depending on the value of M as follows: a) λ-vector of the N-median problem, i.e. $\lambda = (1, \ldots, 1) \in \mathbb{R}^M$; b) λ-vector of the N-center problem, i.e. $\lambda = (0, \ldots, 0, 1) \in \mathbb{R}^M$; c) λ-vector of the $\lfloor M/4 \rfloor$-centrum problems; and d) λ-vector of the (k_1, k_2)-trimmed mean problem, i.e. $\lambda = (0, \ldots, 0, 1, \ldots, 1, 0, \ldots, 0) \in \mathbb{R}^M$ where $k_1 = \lfloor 0.2M \rfloor$, $k_2 = \lfloor 0.2M \rfloor$. 4) *Demand of facilities:* Integer

and uniform in $[10, 20]$. 5) *Capacity of suppliers:* Integer uniformly distributed in $[1.1\frac{\sum_{i=1}^{M} a_i}{N}, 1.4\frac{\sum_{i=1}^{M} a_i}{N}]$. 6) *Transportation cost:* Free self service and integer costs. The values c_{ij}, $i \neq j$, are drawn uniformly in $[0, 200]$.

We solve 5 instances for each possible combination of levels and we report the average and maximum running time, the gap at the root node and number of nodes in the branch and bound tree for each formulation. All computational studies have been performed on a PC with a *Pentium IV* processor with 2.0 GHz and 1 RAM GB. To solve the different instances of the problems we have used XPRESS-IVE solver version 1.17.04, with a code implemented in XPRESS-MOSEL version 1.6.2.

Table 1. Numerical comparison of the three formulations for the median and center problems

		MEDIAN						CENTER					
	M	10		20		30		10		20		30	
	N	3	5	5	10	7	15	3	5	5	10	7	15
3-indexes	$A-CPU$	1.75	4.93					101.26	10881*				
	$M-CPU$	3	6.75					176.3	54110				
	$A-Nod$	158	672					127201	9727686				
	$M-Nod$	263	1181					281546	48200555				
Coverage	$A-GAP$	3.27	9.71	6.11	43.75	6.20	34.07	199.10	310.63	204.26	331.03	216.39	287.84
	$M-GAP$	19.27	19.15	9.08	70.73	7.78	42.45	287.30	367.77	257.47	580.85	236.02	481.39
	$A-CPU$	0.27	7.25	16.58	1838.79	15.26	74.9	47.89	153.09	187.33	1143.51	857.61	3055.98
	$M-CPU$	0.84	18.47	27.21	114.98	96.27	3645.74	36.88	251.94	359.14	3794.84	1444.89	3639.27
	$A-Nod$	13	12.6	373	1398	825	6320	56	169	889	15084	1677	2434
	$M-Nod$	65	47	841	3461	1661	12720	525	701	1827	45664	3189	2889
O-T	$A-CPU$	0.11	0.12	1.49	3.03	32.57	325.18	0.19	0.14	15.69	9.76	539.95	821.5
	$M-CPU$	0,16	0.13	2.1	4.43	39.49	1160.5	0.23	0.18	29.3	20.07	1587.5	3839.2*
	$A-Nod$	18	22	711	2099	11266	120164	73	43	13072	6394	285710	364258*
	$M-Nod$	45	31	1145	23231	13631	407897	89	83	30337	14111	909365	1743737

Table 1 is devoted to the results for the median and center problems and 2 to the k-centrum and trimmed-mean problems. The columns show the results for the different sizes of M and N. Each table has three blocks of rows. The first block refers to the formulation 3-*indexes*, the second block to the new formulation based on covering (1)-(6) (*coverage*) and the last one to the model *O-T*.

Within each block of rows we report on the GAP at the root node, CPU-time to solve the integer problems and number of $NODES$ in the branch-and-bound tree (average and maximum). Empty blocks mean that the solver was unable to find an optimal solution of the model; asterisks mean that some of the problems were not solved to optimality within the 2 hours time limit.

Next, we analyze the results of the experiments. We first observe that formulations 3-*indexes* and *O-T* always provide a zero value to the linear relaxation of the problem. Thus, we do not report on their GAP. Quality of the lower bounds provided by *coverage* depend on the

Table 2. Numerical comparison of the three formulations for the k-centrum and trimmed-mean problems

		k-CENTRUM						TRIMMED-MEAN					
	M	10		20		30		10		20		30	
	N	3	5	5	10	7	15	3	5	5	10	7	15
3-indexes	$A-CPU$	8.25	15.17					13.51	7.98				
	$M-CPU$	16.73	37.55					25.5	13.12				
	$A-Nod$	1193.8	3419					1463	1227				
	$M-Nod$	3197	7967					2787	2139				
Coverage	$A-GAP$	30.84	66.42	39.28	59.29	38.89	64.73	36.93	55.02	35.20	39.47	35.76	44.78
	$M-GAP$	38.51	109.72	52.57	81.36	44.11	94.44	48.51	95.26	37.73	50.08	40.48	63.93
	$A-CPU$	18.23	97.83	117.89	213.93	1571.93	1850.49	7.72	29.32	83.19	28.91	306.58	63.84
	$M-CPU$	41.12	245.95	345.86	521.56	2653.49	3433.25	11.26	70.25	106.68	39.93	557.88	143.96
	$A-Nod$	189	273	523	1418	3684	22162	34	55	238	63	469	244
	$M-Nod$	267	791	1723	3681	6649	79621	91	172	341	93	1003	422
O-T	$A-CPU$	0.14	0.16	2.20	2.57	105.85	318.47						
	$M-CPU$	0.22	0.25	3.84	5.4	272.56	537.48						
	$A-Nod$	50	96	1027	1517	30135	109642						
	$M-Nod$	85	251	2043	3297	80267	177577						

type of problem being rather good for median problems (least than 7% in average), reasonable for k-centrum and trimmed-mean (least than 50%) and poor for the center (above 100%).

The computational experiments also show that 3-*indexes* formulation is far behind the other two formulations in all respects. Overall, model O-T outperforms the others, whenever it is applicable. Nevertheless, when the size of the problem increases *coverage* and O-T tend to perform similarly. This can be observed for the problems with $M = 30$ and it was clearer for larger instances. In addition, one can strengthen formulation *coverage* by using the valid inequalities in Section 2.

References

1. N. Boland, P. Domínguez-Marín, S. Nickel, and J. Puerto. Exact procedures for solving the discrete ordered median problem. *Computers and Operations Research*, 33:3270–3300, 2006.
2. J. Kalcsics, S. Nickel, J. Puerto, and A.M. Rodríguez-Chía. Flexible discrete location problems with capacity constraints. *Submitted*, xx:xx–xx, 2006.
3. A. Marín, S. Nickel, J. Puerto, and S. Velten. A flexible model and efficient solution strategies for discrete location problems. *Submitted*, xx:xx–xx, 2006.
4. S. Nickel. Discrete ordered weber problems. In *Operations Research Proceedings 2000*, 71–76. Springer Verlag, 2001.
5. S. Nickel and J. Puerto. *Facility Location - A Unified Approach*. Springer Verlag, 2005.
6. W. Ogryczak and A. Tamir. Minimizing the sum of the k largest functions in linear time. *Information Processing Letters*, 85(3), 2003.
7. J. Puerto. The radius approach to the capacitated discrete ordered median problem. Presented in *CORAL 2006*. Pto de la Cruz, Spain, 2006.

Part VII

Econometrics, Game Theory and Mathematical Economics

Investment Timing Problem Under Tax Allowances: The Case of Special Economic Zones

Vadim Arkin[1], Alexander Slastnikov[1], and Svetlana Arkina[2]

[1] Central Economics and Mathematics Institute, Nakhimovskii pr. 47, Moscow, Russia. arkin@cemi.rssi.ru, slast@cemi.rssi.ru
[2] University Paris I. svetlana.arkina@malix.univ-paris1.fr

Special economic zones (SEZ) are aimed at creating a favorable environment (in terms of taxation, custom rights and administrative burden) for businesses. Particularly, the SEZ are dedicated to the development of certain types of industries or to the revitalization of several economically depressed areas. The institutional role of SEZ in economic development is largely referenced in the literature (see for example [3]). Tax exemptions are one of the most used stimuli in order to attract investment in SEZ.

In this paper we build a model of investment waiting which describes the behavior of an investor who wishes to invest in a project of creation of a new enterprise in a SEZ. The model takes into account the uncertainty of the cash flows from the future enterprise. Those cash flows are generated by the stochastic dynamics of market prices on goods produced and resources consumed. In our model we will take into account the following tax exemptions, which are common in SEZ of numerous countries: tax holidays on enterprises property tax, accelerated depreciation, reduced rates of corporate profit tax and of unified social tax. The optimal rule of choice of investment timing and its analytical dependence on tax exemptions introduced below and other parameters of the tax system, are obtained. Based upon the example of Russian SEZ, we conduct a modelling analysis of the potential impact of the introduction of those tax exemptions on tax revenues in regional and federal budgets, from newly created enterprises.

1 Investment Waiting Model in Special Economic Zones

As object of investment, we will consider a project of creating a new enterprise in a SEZ. Investment necessary to the realization of the project is supposed to be instantaneous and irreversible. We also assume that, at any time, the investor can either accept the project and start with the investment, or delay the decision until he obtains new information about its environment (prices, demand, etc.). The starting point of this model is the real option theory (McDonald-Siegel model, see [4],[2]).

Let us suppose that the investment in the project starts at moment τ, the cost of necessary investment (without VAT) is I_τ.

At time $\tau + t,\ t \geq 0$ the before-tax profit of the firm is equal to

$$(1 + \gamma_{va})\pi^\tau_{\tau+t} - S^\tau_{\tau+t}, \tag{1}$$

where $\pi^\tau_{\tau+t}$ is value added (the difference between income and material costs without VAT), γ_{va} is the VAT rate, and $S^\tau_{\tau+t}$ is payroll cost.

Taxes, which are paid by the firm, consist of the following:

- value added tax $\gamma_{va}\pi^\tau_{\tau+t}$;
- payroll tax (called in Russia "unified social tax") $\gamma_s S^\tau_{\tau+t}$ where γ_s is the relevant tax rate;
- property (or asset) tax $P^\tau_{\tau+t}$ whose base is the residual cost of assets;
- corporate profit tax (with the rate γ_i) which base is profit (1) minus VAT, depreciation charges $D^\tau_{\tau+t}$ and other costs, including both payroll tax and asset tax.[3]

We divide all assets into two aggregated parts: one of them ("active" part) refers to machinery, tools, equipment etc.(its share in the balance costs of all assets will be denoted as $\psi,\ 0 \leq \psi \leq 1$); and the other ("inactive" part) refers to buildings and structures, whose useful lifetime is long enough.

Depreciation charges at time $\tau + t$ for the project started at τ will be $D^\tau_{\tau+t}{=}\psi I_\tau a_t{+}(1{-}\psi)I_\tau b_t, t \geq 0$, where $(a_t, t \geq 0),\ (b_t,\ t \geq 0)$ are the depreciation "densities" of of active and inactive parts of assets such that $a_t, b_t \geq 0,\ \int_0^\infty a_t\,dt = \int_0^\infty b_t\,dt = 1$.

Since the economic environment can be subject to the influence of various random factors (uncertainty in market prices, demand, etc.), we will consider that the cost of required investment $(I_t, t{\geq}0)$ is a random process, and the value added $(\pi^\tau_{\tau+t}, t{\geq}0)$ is modeled by a family (in

[3] Actually, the corporate income tax equals zero if its tax base (1) is negative. Nevertheless, we shall write the term (1) even if it is negative. This can be viewed as an approximation of the principle of losses carry forward (like deductions from tax base in the future)

$\tau \geq 0$) of random processes, given on some probability space $(\Omega, \mathbb{F}, \mathbf{P})$ with the flow of σ-fields $\mathcal{F}=(\mathcal{F}_t, t \geq 0)$ (the observable information about the system), and random processes are assumed to be $\mathcal{F}$-adapted.

In SEZ, firms can be subject to tax on property tax, and let ν be the length of the period of time during which property tax is not levied.

The expected present value of the firm can be expressed as

$$V_\tau = \mathbf{E}\left(\int_0^\nu [(1-\gamma_i)(Z^\tau_{\tau+t}+P^\tau_{\tau+t})+D^\tau_{\tau+t}]e^{-\rho t}dt + \int_\nu^\infty [(1-\gamma_i)Z^\tau_{\tau+t}+D^\tau_{\tau+t}]e^{-\rho t}dt \,\middle|\, \mathcal{F}_\tau \right), \quad (2)$$

where $Z^\tau_{\tau+t}=\pi^\tau_{\tau+t}-(1+\gamma_s)S^\tau_{\tau+t}-P^\tau_{\tau+t}-D^\tau_{\tau+t}$ is the tax base for corporate profit tax, and ρ is the discount rate.

Optimal timing problem

The behavior of the investor is supposed to be rational in the sense that he chooses the time for investment τ (investment rule), in order to maximize his expected net present value (NPV):

$$\mathbf{E}\,(V_\tau - I_\tau)\, e^{-\rho\tau} \to \max_\tau, \quad (3)$$

where the maximum is considered over all Markov times τ .

Tax payments from the firm are splitted each year between both regional and federal budgetary levels. Hence, federal budget receives VAT, a part of the UST (at the rate γ_s^f), and the federal part of the corporate profit tax (at rate γ_i^f out of γ_i). Regional budget gets enterprise property tax, personal income tax and the regional part of corporate profit tax (at the rate $\gamma_i^r=\gamma_i-\gamma_i^f$). Moreover, as far as tax entries into the budget are concerned, we will take into account the personal income tax $\gamma_{pi}S^\tau_{\tau+t}$ (where γ_{pi} is the relevant tax rate). Simultaneously one can calculate (using formula (2)), the present tax revenue into federal T^f_τ and regional T^r_τ budgets from the firm after investment.

Main assumptions

The amount of *required investment* I_t is described by geometric Brownian motion $I_t=I_0+\int_0^t I_s(\alpha_1 ds+\sigma_1 dw^1_s)$, $t \geq 0$, where $(w^1_t, t \geq 0)$ is a Wiener process, α_1 and σ_1 are real numbers, and I_0 is a given initial

state of the process. The dynamics of *value added* $\pi^\tau_{\tau+t},\ t \geq 0$ is specified by a family of stochastic equations $\pi^\tau_{\tau+t}=\pi_\tau+\int_\tau^t \pi^\tau_s(\alpha_2 ds+\sigma_2 dw^2_s),\ t \geq 0$, where π_τ is $\mathcal{F}_\tau$-measurable random variable, $(w^2_t,\ t\geq 0)$ is a Wiener process, α_2 and σ_2 are real numbers. The pair (w^1_t, w^2_t) is two-dimensional Wiener process with correlation r. We assume that at any moment τ, observing the current prices on both input and output production one can calculate $\pi_\tau = \pi^\tau_\tau$, which is the value added at the "initial moment" of creation of firm, and, hence, can evaluate the future profits from the project before the actual creation of the firm. We suppose that the process π_τ is a geometric Brownian motion with parameters $(\alpha_2,\ \sigma_2)$.

The payroll fund $S^\tau_{\tau+t}$ is supposed to be proportional to the value added , i.e. $S^\tau_{\tau+t} = \mu\pi^\tau_{\tau+t}$, where μ is a given constant ("labor intensity", wage per unit of value added). Such a hypothesis is in accord with the principle of dependence between wages and production activity.

2 Optimal Investment Time and Present Tax Revenues

The optimal timing problem (3) faced by the investor is an optimal stopping problem for the two-dimensional stochastic process (π_t, I_t) with the reward function defined by formulas (3), (2).

Let β be a positive root of the quadratic equation $\frac{1}{2}\tilde{\sigma}^2\beta(\beta-1)+(\alpha_2-\alpha_1)\beta-(\rho-\alpha_1)=0$, where $\tilde{\sigma}^2=\sigma_1^2-2r\sigma_1\sigma_2+\sigma_2^2$ is "total" volatility of the investment project.

The following theorem specifies an optimal rule for investing.

Theorem 1. *Let the amount of required investment I_t and value added π_t be described by geometric Brownian motions with parameters (α_1, σ_1) and (α_2, σ_2), respectively. Suppose that $\tilde{\sigma}>0$, $\alpha_2-\frac{1}{2}\sigma_2^2 \geq \alpha_1-\frac{1}{2}\sigma_1^2$, and $\rho>\max(\alpha_1, \alpha_2)$. Then the optimal investment time for the problem (3) is $\tau^*=\min\{t \geq 0 : \pi_t \geq p^* I_t\}$, and the threshold p^* is defined as*

$$p^* = \left[1-\gamma_i K+\frac{(1-\gamma_i)\gamma_p}{\rho}H_\nu\right]\cdot\frac{\rho-\alpha_2}{[1-(1+\gamma_s)\mu](1-\gamma_i)}\cdot\frac{\beta}{\beta-1},$$

where $K = \psi\int_0^\infty a_s e^{-\rho s}ds + (1-\psi)\int_0^\infty b_s e^{-\rho s}ds,$

$$H_\nu = \psi\int_\nu^\infty (e^{-\rho t}-e^{-\rho s})a_s e^{-\rho s}ds + (1-\psi)\int_\nu^\infty (e^{-\rho t}-e^{-\rho s})b_s e^{-\rho s}ds.$$

Knowing the optimal investment time one can find the expected present tax revenues into the budgets at different levels under the optimal behavior of the investor.

Theorem 2. *Under the above conditions the following formulas hold:*

$$\mathbf{E}(V_{\tau^*} - I_{\tau^*})e^{-\rho\tau^*} = I_0 \left[\pi_0/(I_0 p^*)\right]^\beta \left[1-\gamma_i K+(1-\gamma_i)\gamma_p H_\nu/\rho\right]/(\beta-1);$$

$$\mathbf{E}T^f_{\tau^*}e^{-\rho\tau^*} = I_0 \left[\pi_0/(I_0 p^*)\right]^\beta \left[\gamma^f p^* - \gamma^f_i \left(K + \gamma_p H_\nu/\rho\right)\right];$$

$$\mathbf{E}T^r_{\tau^*}e^{-\rho\tau^*} = I_0 \left[\pi_0/(I_0 p^*)\right]^\beta \left[\gamma^r p^* - \gamma^f_i K + (1-\gamma^r_i)\gamma_p H_\nu/\rho\right],$$

where p^* *is defined in Theorem 1,* $\gamma^r = \{\gamma_{pi}\mu + (1-(1+\gamma_s)\mu)\gamma^r_i\}/(\rho-\alpha_2)$ *and* $\gamma^f = \{\gamma_{va} + \gamma^f_s\mu + [1-(1+\gamma_s)\mu]\gamma^f_i\}/(\rho-\alpha_2)$.

These formulas can be derived similarly to those in [1] (for simpler model).

3 Budgetary Effects of the Creation of New Enterprises: The Example of Russian SEZ

The main results of this paper are related to the study of the dependence of present tax revenues in federal and regional budgets on the duration of tax holidays on property tax. Let us emphasize upon the twofold influence of tax exemptions on budgets. On the one hand, tax entries in budgets are reduced. On the other hand, investment occurs earlier, and can lead to an increase of present amount of taxes levied.

The above presented formulas have been applied to the analysis of currently existing SEZ in Russia.

Starting from 2006, it is possible to create three types of SEZ on the Russian territory: industrial (ISEZ) and technological, or technical-innovation, (TSEZ) and recreational (RSEZ). Zone residents are offered a wide range of benefits (including administrative, customs and tax benefits). In particular, enterprises in SEZ have the 5-years exemption from property and land taxes. Beside that a reduced rate of unified social tax has been introduced in TSEZ (14% instead of 26%). In ISEZ, an increase of the depreciation coefficient is allowed (not exceeding twice the standard rate), and the 30% limit of losses carry forward on future tax periods is cancelled. Local authorities can also cut the corporate profit tax rate to 20% (exerting their right to decrease this rate by up to 4%).

It is shown that the revenues of the federal budget from the created enterprise raise with an increase of tax holidays for both types of zones. ISEZ always provide better revenues in federal budget (as compared with the standard, outside SEZ, taxation system). This increase can

reach 10-15%, but decreases with a rise of volatility of the project. For enterprises located in TSEZ, tax entries in federal budget can be lower than under the standard tax regime. Such situation arises for projects with high volatility and "moderate" labour intensity (under existence of 5-years property tax holidays). However the decrease of fiscal entries is very low and never exceeds 1-2%.

The dependence of tax revenues in regional budget on the length of tax holidays on property tax is more complex, labour intensity μ (wage per unit of value added) plays a substantial role. There are two threshold values μ_1 and μ_2 (which depend on parameters of the project and type of SEZ) which characterize three types of dependencies of tax revenues in regional budget on tax holidays. When $\mu<\mu_1$ the present tax revenues decrease monotonically with the increase of tax holidays; and when $\mu>\mu_2$ – increase monotonically with the increase of tax holidays. When $\mu_1<\mu\leq\mu_2$, we observe a "Laffer effect": present revenues in regional budget increase initially with the rise of tax holidays, then decrease. However, under the 5-years property tax holidays in SEZ, those tax entries can either be higher or lower than under the standard tax regime. This depends on parameters of the project, among which volatility and technical performance. As calculation shown, the increase of tax revenues in regional budget as compared to the standard tax regime, is generally observed when the values of labour intensity are relatively high and the corresponding "threshold value" is lower for TSEZ than for ISEZ.

Thus, the system of tax exemptions for SEZ currently applied in Russia could be efficient (as far as tax entries in federal and regional budgets are considered) only for investment projects with relatively high level of labour intensity and moderate volatility.

This work is supported by RFH (project 07–02–00166).

References

1. Arkin V, Slastnikov A (2004) Optimal stopping problem and investment models. In: Dynamic Stochastic Optimization. Lecture Notes in Economics and Mathematical Systems 532: 83–98
2. Dixit AK, Pindyck RS (1994) Investment under Uncertainty. Princeton University Press, Princeton
3. Litwack J, Qian Y (1998) Balanced or Unbalanced Development: Special Economic Zones as Catalysts for Transition. Journal of Comparative Economics 26 (1): 1–25
4. McDonald R, Siegel D (1986) The value of waiting to invest. Quarterly Journal of Economics 101: 707–727

Computing the Value of Information in Quadratic Stochastic Decision Problems

Sigifredo Laengle

Universidad de Chile, Diagonal Paraguay 257, 833015 Santiago de Chile
slaengle@fen.uchile.cl

Summary. We present a game in which, if one of the players improves his payoff upon obtaining more information, the other player's payoff worsens in such a way that there is a net social loss due to having more information. How can we ensure this does not occur? The results of this paper are (1) the mathematical expression of the (social) value of information in a quadratic non-cooperative game, and (2) the conditions that ensure the social value of information is non-negative.

1 Introduction

This article poses the question of the amount a decision-maker is willing to pay to increase the *quantity of available information* and thus improve his decision. The answer will depend on which of two basic contexts are under consideration: a decision-maker who has no interaction with other players, or one who does have strategic interaction (typically a non-cooperative game). In general terms, the value of information is always non-negative for a decision-maker without interaction, but in a game situation it can be negative. This article presents the non-negativity conditions for the value of information in the case of a game with quadratic cost functions.

There are many examples in the literature of non-cooperative games in which players prefer not to have additional information in order to improve their payoff (see for example [7], [10], [4]) for the general Bayesian games, and [2], [8] for a non-cooperative transportation network). We present a game in which, if one of the players improves his payoff upon obtaining more information, the other player's payoff worsens in such a way that there is a *net social loss* due to having

more information. How can we ensure this does not occur? What is the expression of the information value?

2 An Example of Negative Information Value

Consider two players who are attempting to approximate a common objective[1] denoted ω. The loss function of Player t ($t \in \{1, 2\}$) is given by the distance between this objective and $\overline{x}_t$, his best approximation. However, each player's approximation can be affected by the action $m_{tr}x_r$ of the other player. The effect $m_{tr}x_r$ of Player r on Player t may be interpreted as the action taken by Player r to disturb Player t's approximation (i.e. x_r). If ω is a random variable, the loss function of Player t is expressed as

$$f_t(x_t, x_r) := \frac{1}{2}\mathbf{E}(x_t + m_{tr}x_r - \omega)^2 \quad \text{with } r \neq t.$$

A large class of games of this type may be found in the literature. For example, the equilibrium solution to the classic Cournot duopoly (Fudenberg and Tirole, 1991, [3], p. 215ss.) can be restated as an approximation game in which each player must approximate the demand of a single homogeneous good. In Hinich and Enelow's spatial voting theory (1984, [5]) the authors model voters' objective function as an approximation of their ideal policy. Pursuit-evasion (Isaacs, [6], p. 67ss.) can also be modeled as an approximation game of this type.

To illustrate the conditions that ensure a non-negative information value, we consider three cases of static Bayesian games. In the first case (Case A), both players only have common information on ω. In Case B, one of the players has an additional observation ξ containing information on ω. Finally, in Case C both players have the same additional observation. Case B is thus an asymmetric information Bayesian game, whereas cases A and C are Bayesian games with symmetric information.

To determine the three games' respective equilibria we begin by specifying more precisely certain items introduced above. The variable ω is a real random variable that has an *a priori* normal density function with mean $\overline{\omega}$ and standard deviation σ_ω. This information is common to both players in all cases. If $\overline{x}_t$ is the best approximation of Player t, then, since each player knows that the opposing player (r) will play

[1] For simplicity we temporarily assume that the players have common objectives; later, we will solve the problem assuming that the objectives are not necessarily the same.

$\overline{x}_r$, the equilibrium solution is given by the system of equations $\overline{x}_1 = \mathbf{E}(\omega - \overline{x}_2)$ and $\overline{x}_2 = \mathbf{E}(\omega - \overline{x}_1)$. Solving this system yields the following loss functions[2]

$$f_1(\overline{x}_1, \overline{x}_2) = f_2(\overline{x}_1, \overline{x}_2) = \frac{1}{2}\sigma_\omega^2.$$

In Case B, Player 1 observes the real random variable ξ. We assume that the marginal density of observation ξ is also normal, with mean ω and standard deviation σ_ξ. Both players in this instance know that Player 1 has the additional observation ξ. The equilibrium solution is then given by the system of equations $\overline{x}_1 = \mathbf{E}(\omega - \overline{x}_2|\xi)$ and $\overline{x}_2 = \mathbf{E}(\omega - \overline{x}_1)$, whose result is

$$f_1(\overline{x}_1, \overline{x}_2) = \frac{1}{2}\frac{\sigma_\omega^2}{\sigma_\omega^2 + \sigma_\xi^2}\sigma_\xi^2 \quad \text{and}$$

$$f_2(\overline{x}_1, \overline{x}_2) = \frac{1}{2}\frac{\sigma_\omega^2}{\sigma_\omega^2 + \sigma_\xi^2}(\sigma_\xi^2 + \sigma_\omega^2(m_{21} - 1)^2).$$

Case C, in which both players observe ξ, is solved in analogous fashion with the following result

$$f_1(\overline{x}_1, \overline{x}_2) = f_2(\overline{x}_1, \overline{x}_2) = \frac{1}{2}\frac{\sigma_\omega^2}{\sigma_\omega^2 + \sigma_\xi^2}\sigma_\xi^2.$$

Having set out the foregoing results we now propose the following game. Assume that each player has the option of using the additional observation ξ and knows whether or not the opposing player is using the same information. As before, both players play simultaneously. The gain from using the information obviously depends on whether or not the other player is also using it. Thus, Player 1 can opt to not use the information (Decision 1) or to use it (Decision 2), and Player 2 must also decide whether to not use it (Decision I) or to use it (Decision II). The gain or loss to Player t from using the information is obtained as the difference between the respective values of the cost function for the equilibrium strategies with and without the additional information. If, for example[3], $m_{12} = m_{21} = \sqrt{3} + 1$ and we define the positive number

$$a := \frac{1}{2}\sigma_\omega^2\frac{\sigma_\omega^2}{\sigma_\omega^2 + \sigma_\xi^2},$$

then the gain matrix of the game is as shown in Table 1.

[2] In order to calculate this results, certain algebraic manipulations were performed using the software program Maple V. The Nash equilibrium solutions $(\overline{x}_t)$ are not presented here as they do not contribute significantly to our analysis.

[3] This value of m_{12} does not limit the generality of the analysis.

Table 1. Information game

	I	II
1	$(0,0)$	$(-2a,a)$
2	$(a,-2a)$	(a,a)

Observe that this game has a pure strategy Nash equilibrium that consists in both players preferring to use the additional information. In other words, the Nash equilibrium is $(2,\mathrm{II})$. Recall that this result is obtained if $m_{12} = m_{21} = \sqrt{3}+1$ with any value of the mean $\overline{\omega}$.

In this work we are interested in the problem represented in Table 1. Assume, then, that Player 2 cannot use the available information or that the observation is too expensive for him to obtain it. In this situation, Player 1 will use the available information to achieve a decrease a in his costs while bringing about an increase $2a$ in Player 2's costs. In other words, while Player 1 gains unilaterally, Player 2 loses twice what Player 1 gains. But of particular significance is that Player 1's unilateral decision has a social cost equal to a. How can we ensure that this does not occur? What conditions must be imposed on the information available to the players and/or the interaction between them to ensure the social benefits of the information are non-negative?

3 The Generalized Quadratic Stochastic Game

In a more general context, let us define following information structures. Let $(\Omega,\mathcal{B},\mathbf{P})$ be the information structure of the game. It is a probability space defined by the set Ω, a σ-algebra $\mathcal{B}$ defined on Ω and the probability measure $\mathbf{P}$. The available information structure to each player can be modeled as $(\Omega,\mathcal{C},\mathbf{P})$, where $\mathcal{C}$ is a sub-σ-algebra of $\mathcal{B}$ (i.e. $\mathcal{C}\subset\mathcal{B}$). For example, the sub-$\sigma$-algebra corresponding to both players in Case A of the previous example is $\mathcal{B}(\Re)\otimes\{\emptyset,\Re\}$, where $\mathcal{B}(\Re)$ are the Borel subset of $\Re$. In Cases B and C, the sub-σ-algebra of players that have the information ξ is $\mathcal{B}$, i.e. they have all the information of the game.

With the above definitions, we define the **event space** as $V := \mathcal{L}^2(\Omega,\mathcal{B},\mathbf{P})$, that is, the set of all random variable with finite variance. This space, endowed with a scalar product $\langle v,w\rangle := \mathbf{E}(vw)$ for all $v,w\in V$, is a Hilbert space, and $|\cdot| := \langle\cdot,\cdot\rangle^{1/2}$ is a norm on V. It is known, that the set $\mathcal{L}^2(\Omega,\mathcal{C},\mathbf{P})$, for any $\mathcal{C}\subset\mathcal{B}$, is a closed subspace of V. With the help of this abstraction level, it is easier to obtain the main results of this paper.

The generalized quadratic bayesian game is defined as follows: There are two players, each of whom chooses a decision from his respective closed subspaces $E_1 \subset V$ and $E_2 \subset V$. The cost function for each player t is

$$f_t(x_1, x_2) := \frac{1}{2}|x_t + M_{tr}x_r - u_t|^2 \quad \text{with } r \neq t,$$

where M_{rt} is a bounded linear operator from V to V (we write $M_{rt} \in L(V)$) for $\{r, t\} \in \{1, 2\}$. The problem (Q_2) is then expressed as

(Q_2) Find $\overline{x}_1 \in E_1$ and $\overline{x}_2 \in E_2$, such that

$$f_1(\overline{x}_1, \overline{x}_2) = \min\{f_1(x_1, \overline{x}_2) : x_1 \in E_1\}$$
$$f_2(\overline{x}_1, \overline{x}_2) = \min\{f_2(\overline{x}_1, x_2) : x_2 \in E_2\}$$

Before continuing, we require certain definitions given that we are working in $V \times V$ space. Let M be defined as $\{Mv\}_t := v_t + M_{tr}v_r$ for every $v \in V \times V$ and $v = \{v_t, v_r\}$. Assume that M is an isomorphism over[4] V. We also define M_E as $\{P_E M\}_t = v_t + P_{E_t}M_{tr}v_r$, where P_E is the orthogonal projector onto the subspace $E_1 \times E_2$ given by $\{P_E v\}_t = v_t + P_{E_t}v_r$ with $r \neq t$. The problem (Q_2) has a unique solution and a proof is found in Aubin (1979, [1]), while Laengle (2000, [9]) gives the solution for the restricted case on closed subspaces.

Lemma 1 (Nash equilibrium). *The problem (Q_2) has a unique solution. Also, the equilibrium solution is*

$$\overline{x} = M_E^{-1} P_E u.$$

4 Non-negativity Conditions of Information Value

We now define the information value for the problem (Q_2). Let F_1, F_2 be two closed subspaces of V such that $E_1 \subset F_1$ and $E_2 \subset F_2$. Assume also that $\overline{y}_1, \overline{y}_2$ are the Nash equilibrium solutions in the subsets F_1, F_2. We define the **information value of the problem Q_2 of F with respect to** E as

$$I_2(E, F) := \underbrace{f_1(\overline{x}_1, \overline{x}_2) + f_2(\overline{x}_1, \overline{x}_2)}_{\text{Solution in } E} - \underbrace{(f_1(\overline{y}_1, \overline{y}_2) + f_2(\overline{y}_1, \overline{y}_2))}_{\text{Solution in } F}.$$

[4] It can be shown that this condition is equivalent to operator M being *bounded below*, that is, $\langle Mv, v\rangle \geq |v|^2$ for every $v \in V$.

In what follows, we prove that the condition which must be imposed on the interaction operator M to ensure the value of information for the game is non-negative is that the subspaces E and F must be invariant under the operator M, that is, $M(E) \subset E$ and $M(F) \subset F$. This invariance condition may relate to the information symmetry of the interactions if the operator m is a constant multiplier: *If all players' strategies are observable by each player, the game's information value is non-negative.*

The following theorem demonstrates that the non-negativity condition of the value of information is that the observation subspaces E and F are invariant under operator M. The proof of this theorem requires certain intermediate results found in [9].

Theorem 1 (Non-negativity of I_2, [9]). *If the interaction operator M is an isomorphism and E, F are invariant subspaces under M, then the game's information value is non-negative, that is, $I_2(E, F) \geq 0$.*

References

1. J. Aubin. *Mathematical Methods of Game and Economic Theory.* North-Holland, 1979.
2. Bean, N.G., Kelly, F.P., and Taylor, P.G. Braess's paradox in a loss network. *Journal of Applied Probability*, 34:155–159, 1997.
3. D. Fudenberg and J. Tirole. *Game Theory.* The MIT Press, 1991.
4. O. Gossner. Comparison of information structures. *Games and Economic Behavior*, 30:44–63, 2000.
5. Hinich, M. and Enelow, J. *The spatial theory of voting: an introduction.* Cambridge University Press, 1984.
6. R. Isaacs. *Differential Games.* John Wiley & Sons, 1965.
7. Kamien, M. I., Y. Tauman, and S. Zamir. On the value of information in a strategic conflict. *Games and Economic Behavior*, 2:129–153, 1990.
8. Korilis, Y.A., Lazar, A.A., and Orda, A. Avoiding the Braess paradox in non-cooperative networks. *Journal of Applied Probability*, 36:211–222, 1999.
9. S. Laengle. *Evaluierung von Alternativen Informationsstrukturen bei Stochastischen Entscheidungsproblemen.* PhD thesis, Universitaet Konstanz, Universitaetsstr. 10, 78464 Konstanz, Germany, 2000. (in german).
10. A. Neyman. The positive value of information. *Games and Economic Behavior*, 3:350–355, 1991.

How Often Are You Decisive: an Enquiry About the Pivotality of Voting Rules

Tobias Lindner

Institute for Economic Theory and Operations Research, University of Karlsruhe, Germany
lindner@wior.uni-karlsruhe.de

Summary. The probability that an individual is decisive in an election is an important criterion for the evaluation of voting rules. It depends on factors such as the number and the behaviour of the other voters, the available alternatives and, of course, also on the voting rule itself. Classical power indices like the Banzhaf- or the Shapley-Shubik-Index are only applicable in special cases. In this paper, an approach is proposed that can be viewed as a natural extension of the Banzhaf-Index for more than two alternatives and for different stochastic assumptions.

The approach is applied to plurality voting with two and more alternatives and computed for variations of the number of voters and alternatives. For three alternatives the pivotality is also computed for the Borda-Rule. The comparison of the computations for three alternatives shows that the probability of being decisive under the Borda-Rule is uniformly larger than under plurality voting.

1 Introduction

The influence of a participant in a decision or voting procedure is an important criterion for many aspects regarding the evaluation of voting rules: On the one hand, a voter can use his influence to manipulate the result of an election, i.e. she can give a vote that does not correspond to her true preferences over the alternatives. The well-known Gibbard-Satthernwaite Theorem states that every social choice function over three or more alternatives can be manipulated by misrepresentation of preferences. An important question is which voting rules have a high chance and which a low chance of being susceptible to strategic manipulation.

On the other hand in democratic decision procedures like parliament elections the participation of many voters is desired, and therefore the

chance that one's vote makes a difference should be high. There is the well known problem of lack of incentive for voters to participate in (large) elections when the probability that they are decisive is low. Also, in decision procedures like committees the probability that a participant has influence on the outcome should be high, in order to induce committee members to carefully consider their decisions.

One way to measure influence is via so-called power indices like the Banzhaf- and the Shapley-Shubik-Index, originally used in cooperative game theory. In fact, these indices are only applicable in special situations with two alternatives and under specific probabilistic assumptions about the behaviour of voters. For example, the probabilistic assumption of the Banzhaf-Index is that voters act independently and vote with 0.5 for the first and 0.5 for the second alternative (see [5]).

In this paper we propose a general approach to measure the influence of an individual in a voting situation called pivotality. By pivotality we mean the probability that one's vote can change the winner of an election, i.e. that one is a pivotal figure in the voting situation. Manipulability is a special case: a voting rule is said to be manipulable, if by *misrepresenting* her preferences a voter can change the result of the election *to her own benefit*.

Pivotality depends on four factors: the number n of voters and the number k of alternatives, the behaviour of the voters (modelled by a discrete probability distribution P over the possible votes) and finally the voting rule itself (denoted by f). Furthermore we assume in what follows that voters act independently.

2 The Model

Let X be a finite set of alternatives (e.g. candidates, policies) and $|X| = k$. Formally, a voting rule is a social choice function $f : V \to X$ which maps a vector $v = (\succ_1, \ldots, \succ_n)$ of the votes of n individuals to a chosen alternative ("the winner") $r \in X$. Each vote is a preference ordering (i.e. a strict linear ordering) of the k alternatives. Note that with k alternatives there are $m = k!$ different votes.

We will only consider voting rules that satisfy anonymity, i.e. we assume $f(v) = f(\sigma(v))$ for all permutations σ of the votes. Thus, the relevant information for determining the result of an election is the anonymous profile, i.e. the frequency distribution of the different votes, which will be denoted by $a = (a_1, \ldots, a_m)$, where a_i is the frequency of votes of the i-th type, where votes ("types") are ordered in some way. The behaviour of a voter is modelled by a discrete probability

distribution $P = (p_1, \ldots, p_m)$ over the possible votes. The probability to observe a particular anonymous profile follows a multinomial distribution with parameters n and P:

$$p(a = (a_1, a_2, \ldots, a_m)) = \binom{n}{a_1, a_2, \ldots, a_m} p_1^{a_1} \; p_2^{a_2} \; \ldots \; p_m^{a_m} \quad (1)$$

To calculate the pivotality of a voting rule in dependence of n, k and P we have to sum the probability of all anonymous profiles at which an additional voter can change the outcome of the voting rule. We denote the set of all anonymous profiles by A and the subset of the pivotal profiles by A^p. This set, in turn, consists of two subsets A_1^p and A_2^p corresponding to two different ways to be pivotal: first, one can influence the election in a way that one's own vote unambiguously decides the winner, for example if there is a tie among the other votes (set A_1^p); alternatively, it is possible that one can only produce a tie and the final outcome is decided by a random device (set A_2^p). The pivotality of every voting rule that satisfies anonymity and where voters act independently (and are identically distributed) is described by:

$$\sum_{a \in A_1^p} \binom{n}{a_1, a_2, \ldots, a_m} p_1^{a_1} \; p_2^{a_2} \; \ldots \; p_m^{a_m} \quad (2)$$

$$+\tfrac{1}{2} \sum_{a \in A_2^p} \binom{n}{a_1, a_2, \ldots, a_m} p_1^{a_1} \; p_2^{a_2} \; \ldots \; p_m^{a_m} \quad (3)$$

The pivotality of a voter consists of two sums: the first is the sum of the probability of those situations, where one can unambiguously decide the winner, and the second is the sum of those situations, where one can produce a tie. The latter is weighted by $\frac{1}{2}$, the probability that the following tie-breaker leads to an alternative outcome[1].

The elements of A^p and thus the value of the expression above were computed using Matlab. First, the elements of A_1^p and A_2^p are determined. Subsequently, the probabilities of their occurrence are computed using the multinomial distribution. Although the runtime of this procedure can be long, the advantage of the approach is that the pivotality is computed exactly instead of estimating it by Monte-Carlo-Simulation (for more details see [3]).

[1] From the viewpoint of pivotality, ties with more than two alternatives are subsumed under the set A_1^p. This is in contrast to the case of manipulability, i.e. the probability of being able to influence the outcome to one's own benefit.

For most voting rules, pivotality is equivalent to the (non-normalized) Banzhaf-Index β', provided that there are only two alternatives with $P = (0.5; 0.5)$. To see this, let Z be a random variable that counts the votes for the first alternative and let s be the number of so-called swing coalitions (i.e. situations with a pivotal voter). If n is even, one is pivotal if and only if $\frac{n}{2}$ individuals are voting for one alternative (there is a tie and one breaks it with one's vote). But these are exactly the situations in which one is the "swing voter." For $p = 0.5$ one obtains the Banzhaf-Index:[2]

$$P\left(Z = \frac{n}{2}\right) = \binom{n}{\frac{n}{2}} \cdot p^{\frac{n}{2}} \cdot (1-p)^{n-\frac{n}{2}} \tag{4}$$

$$= \binom{n}{\frac{n}{2}} \cdot \left(\frac{1}{2}\right)^{\frac{n}{2}} \cdot \left(\frac{1}{2}\right)^{n-\frac{n}{2}} \tag{5}$$

$$= \frac{s}{2^n} = \beta' \tag{6}$$

The expression in (4) has been first described by [4] and [1] and can be approximated by Stirling's Formula for large n.

3 Applications

We will use now our general approach to measure pivotality in three examples; in each pivotality is calculated under a variation of one of the four influencing factors.

3.1 Comparison of Stochastic Assumptions

As already noted in the introduction, the Banzhaf-Index is only applicable for special cases. We examine the pivotality of a simple election with two alternatives under majority rule by computing the values of (4) for $p \in (0; 1)$ and $n = 2, 3, \ldots, 100$ which is displayed in Figure 1. The result is "knife-edged": For $p = 0.5$ the pivotality of this election is just the Banzhaf-Index; even for small deviations from $p = 0.5$ the pivotality is declining rapidly (see also [2]).

One important application is to determine for a given level of pivotality the appropriate groupsize of voters, if p is known or can be estimated reliably. For this, we have to calculate "iso-pivotality-lines",

[2] If n is odd, then there two situations where one's vote can produce a tie - in each there is a chance of 0.5 to win.

i.e. combinations of p and n with the same pivotality (see Figure 1 right).

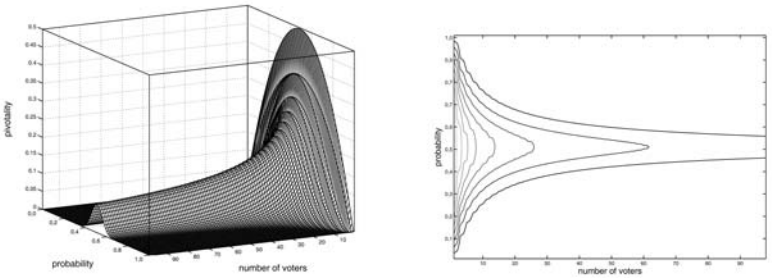


Fig. 1. Pivotality of majority rule with two alternatives

3.2 Comparison of the Quantity of Alternatives

While more voters usually yield less pivotality we can also examine the influence of more alternatives in an election. We calculate the pivotality of the plurality rule for $k = 2, \ldots, 10$ under the impartial culture assumption (i.e. under the assumption that each preference ordering has equal probability for each voter). Therefore, we can compare the

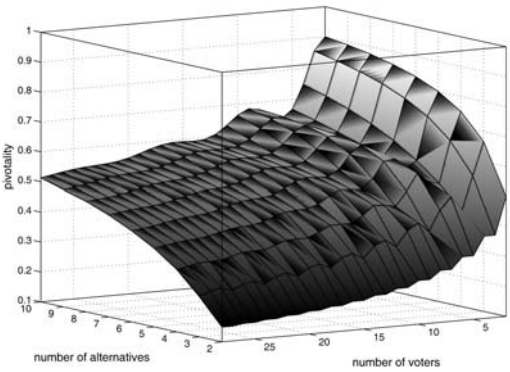


Fig. 2. Pivotality of plurality rule for different quantities of alternatives

pivotality for variations of the number of alternatives. The values are shown in Figure 2: more alternatives give rise to more pivotality under our assumptions.

3.3 Comparison of Voting Rules

Finally, we compare different voting rules in a given situation. We examine the pivotality of the plurality rule and the Borda Count for $n = 2, \ldots, 30$ voters and three alternatives under the impartial culture assumption. The comparison in Figure 3 shows that the pivotality of the Borda Count is uniformly larger than the one of the plurality rule in this situation.

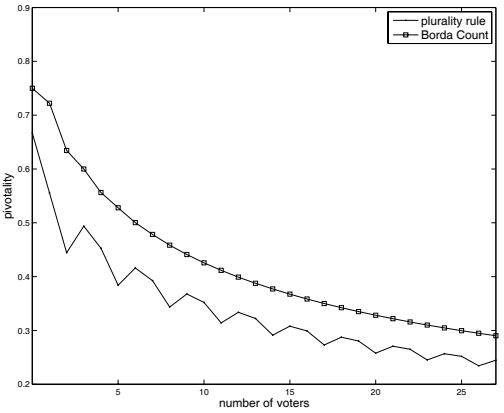


Fig. 3. Comparision of plurality rule and Borda Count

References

1. N. Beck. A note on the probability of a tied election. *Public Choice*, 23(1):75–79, 1975.
2. S. Kaniovski. The exact bias of the banzhaf measure of power when votes are not equiprobable and independent, 2006.
3. T. Lindner. Zur Pivotality von Abstimmungsregeln. Diplomarbeit am Institut für Wirtschaftstheorie und Operations Research, January 2007.
4. L. S. Penrose. The elementary statistics of majority voting. *Journal of the Royal Statistical Society*, 109(1):53–57, 1946.
5. P. D. Straffin. Homogeneity, independence, and power indices. *Public Choice*, 30(1):107–118, 1977.

Part VIII

Energy, Environment and Life Sciences

A System Analysis on PEFC-CGS for a Farm Household

Kiyoshi Dowaki and Takeshi Kawabuchi

Tokyo University of Science, 2641 Yamazaki, Noda, Chiba, Japan 278-8510

Summary. Since 2005, the Japanese government has had great efforts to promote PEFC-CGS (Polymer Electrolyte Fuel Cell Co-generation System) to a household sector in order to mitigate CO_2 emissions. Because of this situation, there is a plan in which 480 units of 1kW PEFC-CGS for a household sector will be installed. However, the heat supply through PEFC-CGS can be often excess energy against heating and/or hot water demands. This is likely to aggravate the cost condition.
On the other hand, in Japan, the number of farm household which possesses green houses is more than 260 thousand points as of 2005. Thus, we focused on the energy demand in which a house and a greenhouse were combined. Using GAMS program, we analyzed the operation condition of PEFC-CGS so as to decrease the excess energy supply, maximizing CO_2 emission reduction and/or the energy cost reduction. On the performance of PEFC-CGS, we calculated the performance model based on electrochemistry and thermodynamics in VBA program. As a result, the power efficiency of 30.3 to 35.2 % and the heat recovery efficiency of 31.6 to 37.3 % were obtained. The part-load operation of PEFC-CGS was considered, too. That is, the part-load operation means that PEFC-CGS shuts down unless the power supply is higher than a minimum electricity demand. Thus, the optimization model on CO_2 emissions or an operational cost is a nonlinear mixed integer model.
Finally, assuming that the total area of a green house was 100 m^2, the maximum reduction of CO_2 emission was 3.46 to 4.06 t-CO_2/yr, compared to the conventional energy supply through fossil fuels origin. Likewise, that of an operational cost was 61,530 to 64,770 yen/yr.

Key words: PEFC-CGS, Farm house, Green house, A nonlinear mixed integer model.

1 Introduction

Since the Kyoto protocol became effective, in Japan, we have tried to promote PV system, gas-engine CGS, PEFC-CGS and the energy saving system (ex. an air conditioner or an advanced boiler) at the time when we build a new house. Especially, Ministry of Economy, Trade and Industry (METI) of Japan has a plan to promote 480 units of 1kW PEFC-CGS into a household sector. However, the heat supply through PEFC-CGS can be often excess energy against heating and/or hot water demands. As a result, this is likely to bring the worse cost condition. On the other hand, the number of farm household which possesses green houses is more than 260 thousand points as of 2005.
Against such a background, we proposed the new PEFC-CGS considering the combination of a farm house and a greenhouse. Since there are four seasons in Japan, the temperature difference between the lowest one and the highest one is likely to be approximately 30 ℃ through the year. That is, the more heat demand would be ensured.
In this paper, we selected the farm houses in Yamaguchi prefecture of Japan as model houses. They cultivate cucumbers and orchids in greenhouses. The energy demands of the house and the greenhouse were estimated by the interviews to farmers, the annual report on the family income and expenditure survey, and the past related data book[1]. We calculated the performance of PEFC-CGS due to VBA program, based on the theories of electrochemistry and thermodynamics[2,3]. Note that the scale of PEFC, whose fuel is town-gas (natural gas), is assumed to be 2kW.
Finally, based on the annual energy demand of a farm house with a green house, and on the performance of PEFC-CGS, we solved the optimization problem on CO_2 emission and/or an operational cost, using GAMS program.

2 Peformance of PEFC-CGS

We set up the parameters of PEFC-CGS and calculated the performance. As a result, the power efficiency was 30.3 to 35.2 % and the heat recovery efficiency was 31.6 to 37.3 % . The output curve for a part-load operation can be approximated as a quadratic function and a monotone function. That is, the net output through PEFC-CGS at i-th month and j-th hour ($FC_p(V)$), the power efficiency ($\eta_e(V)$) and the heat recovery efficiency ($\eta_h(V)$) in each cell voltage can be shown as Eqs. (1)-(3), using the constants evaluated by a least squares method.

Note that the operating voltage is from 0.650 to 0.755 Volts.

$$\mathrm{FC_p(V)} = 40.76\mathrm{V(i,j)}^2 - 64.48\mathrm{V(i,j)} + 25.71 \quad (1)$$

$$\eta_\mathrm{e}(\mathrm{V}) = -0.293\mathrm{V(i,j)}^2 + 0.882\mathrm{V(i,j)} - 0.147 \quad (2)$$

$$\eta_\mathrm{h}(\mathrm{V}) = 0.0147\mathrm{V(i,j)}^2 - 0.559\mathrm{V(i,j)} + 0.730 \quad (3)$$

3 Energy Demand of a Farm House with a Greenhouse

Due to the interviews into a few farmers, we acquired that the growing temperature of cucumbers and that of orchids was 13 ℃ and 20 ℃ respectively. The minimum atmospheric temperature on January drops to 1.0 ℃ in Yamaguchi prefecture. Assuming that the thermal demand for a greenhouse was in proportion to the temperature differences, the specific energy demand of cucumbers was 0.87 MJ/m^2℃ and that of orchids was 0.66 MJ/m^2℃Ȯn the energy demand in a farm house, we estimated it using the past specific energy demand of a household sector and the energy prices of electricity, gas and heating oil in the model area[1]. In this study, the average gross floor area of a farm house in Japan is 95.02 m^2. Assuming that the greenhouse area of cucumbers and/or orchids was constant in 100 m^2, the thermal demand was approximately 2.5 times in comparison to that of a farm house only.

4 System Analysis

Using the specific CO_2 emissions and the energy prices[4], we executed the optimization problem which is a nonlinear mixed integer model.
In the conventional case, the electricity demand is supplied through the commercial power company. Likewise, the heating and hot water demands are supplied through a boiler in use of fossil fuel.
In contrast, in our proposal cases, the energy demand is supplied through PEFC-CGS, and a boiler is used as a backup system. On the surplus power supply through PEFC-CGS, it is assumed that the surplus power can be sold to the commercial electric companies due to the regulation of RPS (Renewable Portfolio System).
Thus, if the annual reduction of an operational cost Red_Cost [yen/yr] and the annual reduction of CO_2 emission and Red_CO_2 [t-CO_2/yr] for the conventional case are shown as the following equations.

$$\max.(\mathrm{Red\_CO_2}), \mathrm{Red\_Cost} \geq 0 \quad \text{or } \max.(\mathrm{Red\_Cost}), \mathrm{Red\_CO_2} \geq 0 \quad (4)$$

Subject to

$$X(i,j)FC_p(V) + Con_p(i,j) = F_e(i,j) + F_h(i,j)/5.4 + G_e(i,j) + \Delta(i,j) \tag{5}$$

$$X(i,j) = 1\ (FC_p(V) \geq SPow),\ X(i,j) = 0\ (FC_p(V) \leq SPow)$$

$$PF(i,j) = X(i,j)FC_p(V)/\eta_e(V) \tag{6}$$

$$F_{hw}(i,j) + G_{hw}(i,j)/(1-\varepsilon) \geq PF(i,j)\eta_h(V) + BF(i,j)\eta_b \tag{7}$$

Where, $Con_p(i,j)$, $F_e(i,j)$, $F_h(i,j)$, $G_e(i,j)$, $\Delta(i,j)$, $F_{hw}(i,j)$, $G_{hw}(i,j)$, $PF(i,j)$ and $BF(i,j)$ are the power supply through PEFC-CGS [kW], the conventional power [kW], the electricity demand of a farm house [kW], the heating demand [MJ/h], the electricity demand of a greenhouse [kW], the salable electricity [kW], the hot water demand of a farm house [MJ/h], that of a green house [MJ/h], the fuel rate of PEFC-CGS [MJ/h] and that of a boiler [MJ/h], respectively. $X(i,j)$ is an integer of 0 or 1. Also, SPow, ε and η_b are the shutdown power [kW](=0.5 kW), the heat trasportation loss (=5%) and the boiler efficiency (=80%). The maximum reduction of CO_2 emission and that of an operational cost compared to the conventional case are shown in Figs. 1 and 2.

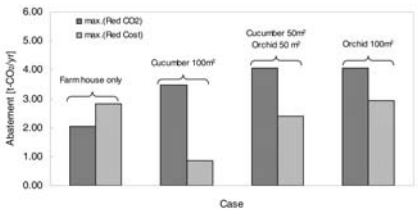


Fig. 1. Max. CO_2 emission reduction

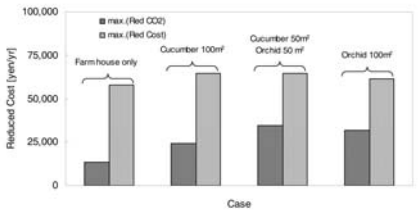


Fig. 2. Max. operational cost reduction

As a result, Red_CO_2 was 3.46 to 4.06 t-CO_2/yr(see Fig. 1). In contrast, Red_Cost was 61,530 to 64,770 yen/yr(see Fig. 2). These results indicate that there are benefits on a house with a greenhouse to some extent, compared to the case of a farm house only. In addition, the excess heat supply through PEFC-CGS, which is usually waste energy, can be decreased significantly.

5 Conclusions

We applied PEFC-CGS for a farm house with a greenhouse in which cucumbers and/or orchids were harvested. Consequently, we concluded that we were able to obtain good benefits on the operational cost reduction and/or CO_2 emission reduction through 2kW PEFC-CGS, compared to the conventional case.

References

1. The specific data on fuel, light and water in a building sector (in Japanese). (1995) Waseda University Press, Japan
2. T. V. Nguyen et al. (1993) A water and heat management model for proton-exchange-membrane fuel cell. J. Electrochem. Soc. Vol. 140. No. 8: 2178-2186
3. J. S. Yi and T. V. Nguyen (1993) An along-the-channel model for proton exchange membrane fuel cell. J. Electrochem. Soc. Vol. 145. No. 4: 1149-1159
4. Handbook of Energy & Economics Statistics in Japan. (2004) The Energy Conservation Center, Japan

Taming Wind Energy with Battery Storage

Andreas T. Ernst and Gaurav Singh

CSIRO Mathematical and Information Sciences, Private Bag 10, South Clayton VIC 3169, Australia. Andreas.Ernst, Gaurav.Singh@csiro.au

1 Background

The use of wind to generate electrical energy is becoming more popular around the world as global efforts are made to deal with green house gas emissions from more traditional sources of energy. In Australia wind energy is one of the technologies being promoted by mandatory renewable energy targets set by the government [1]. Even though wind energy is more economical and eco-friendly it has one significant problem. The electricity production is inherently highly variable and difficult to predict. Over longer time scales it means that it is difficult to match electricity generation to the daily and seasonal patterns of demand. On shorter time scales the higher frequency "noise" in electricity output causes problems for network stability and managing the short term dispatch of generators to meet demand.

Previous work in this area is on battery-management system [2]. Although, we will focus on how to smooth the output over shorter timer periods which has benefits mainly in terms of network stability. The CSIRO is developing a prototype system in Australia that will use specially developed batteries to store some of the wind energy. The aim is to use a relatively small amount of storage in order to smooth the wind output making it more predictable and reliable in the short term. In the Australian market the shortest time scale for bidding and scheduling of generation is a five minute interval. Thus it is important to ensure smoothness and stability over five minute intervals. Figure 1 shows that the wind power output is very variable, and that even for two windfarms in very different locations some of the rise and fall events may be synchronised.

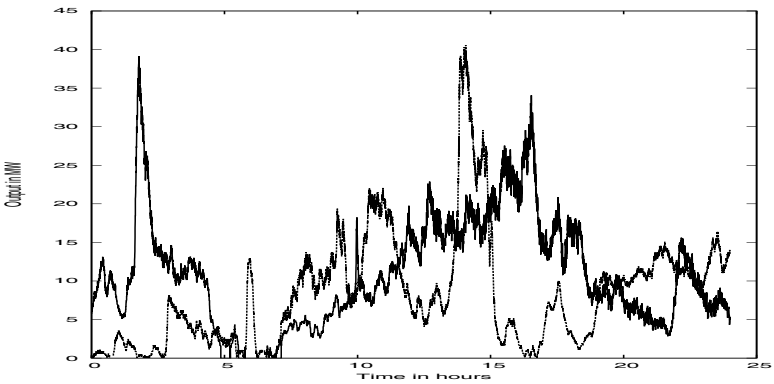


Fig. 1. Example of wind power output for two Australian windfarms

1.1 Formulation

We consider a system where a variable power source is linked to the electricity grid through a device that contains a controller, a battery storage mechanism and an inverter for converting between the AC power of the generator (and as required by the grid) and the DC power of the battery. For the purposes of this paper several assumptions are made: the battery has a fixed capacity; the rate at which it can accept or deliver electrical energy is unrestricted; the controller can instantaneously make a decision on how much power to put into the grid and/or to the battery; and the wind power is only measured at discrete time steps, for example every second.

In this paper the *ramp rate* at time t is defined as the difference between the maximum and minimum power output over the preceding five minutes. Smoothness of output is then defined in terms of a limit on the maximum ramp rate. Let W_t be the instantaneous wind production at time t, C the battery capacity, R the limit on ramp rate and D the number of time steps that make up the five minute ramp rate window. We define decision variables x_t as the power going into the grid during time interval t and s_t as the battery storage at the end of time interval t. We can now define the tactical wind energy optimisation problem (TWEOP), which is to minimise the average ramp rate violation over some fixed time period T:

$$\min \frac{1}{T} \sum_{t \in T} \max \left\{ 0, \max_{i=0,\ldots,D-1} \{x_{t-i}\} - \min_{i=0,\ldots,D-1} \{x_{t-i}\} - R \right\} \quad (1)$$

$$\text{Subject to: } s_t = s_{t-1} + W_t - x_t \quad \forall\, t \in T \quad (2)$$

$$0 \leq s_t \leq C \quad \forall\, t \in T \quad (3)$$

$$x_t \geq 0 \quad \forall\, t \in T \quad (4)$$

The state equation (2) for determining the battery storage level is based on wind input and power output, where for simplicity we have chosen to measure energy in units corresponding to the power multiplied by the time step size.

In principle, this problem be solved as a linear program by introducing additional $O(DT)$ variables and constraints to linearise the objective (1). Though, for big values of T this problem quickly becomes very large. An approximate solution to large TWEOP instances can be obtained through a kind of co-ordinate descent in which a solution is improved by reducing power output at some time t and increasing it at another time t' until either a storage constraint is hit or until neither t nor t' contribute to any ramp rate violations. We refer to this as the *descent* method for TWEOP.

The TWEOP is not particularly useful by itself as it requires exact knowledge of the wind over the whole time period, but it forms the building block for several other methods.

2 Estimating Battery Capacity

A strategic question is how large the battery capacity should be in order to avoid having any ramp rate violations. We can use historical data to obtain estimates of the battery capacity required.

To get a lower bound on the battery capacity required the historical data can be used to solve a modified version of TWEOP in which the battery capacity is a decision variable and the objective is to minimise C. The ramp rate limit then becomes a hard constraint of the form:

$$\max_{i=0,\ldots,D-1} \{x_{t-i}\} - \min_{i=0,\ldots,D-1} \{x_{t-i}\} \leq R. \quad (5)$$

We refer to the (linearised) version of this problem as the capacity minimisation LP (CminLP). Since this problem is too large for big data sets we also consider an approximate version of the problem in which the ramp rate constraint (5) is replaced by the much simpler requirement $-R/D \leq x_t - x_{t-1} \leq R/D$. We refer to this approximation linear program as the ACminLP. Since any feasible solution to ACminLP is

feasible for CminLP, the optimal capacity calculated with ACminLP is at least as large as that obtained from CminLP.

An alternative approach is to start with a control strategy and an arbitrary battery capacity. Repetitive simulations of the operations at a facility can be used to determine the minimum battery capacity required in order to avoid ramp rate violations for a given strategy. This brings us to the question of what is a suitable method for determining how to use the battery in each time interval.

3 On-line Heuristics

We consider three on-line heuristics that differ in the way they manage the battery capacity:

Reactive The reactive heuristic only uses the battery when failing to make use of the facility would cause a ramp rate violation. In that case just enough power is stored or released to stay within the ramp rate limits (or the maximum amount of power possible if the battery limit is reached). At all other times the reactive method uses an exponential decay to bring the battery towards a half-full level so as to maximise the flexibility in either direction.

Proactive: The proactive method tries to manage the battery capacity more actively through a series of rules of thumb. For example it aims to keeping the battery nearly full when the current power output is high as future output is more likely to drop significantly than rise much further.

Low Battery: The essential idea of this method is that for high ramp rates or large batteries it makes sense to keep the battery level as low as possible at all times while always outputing at most R with any peaks being absorbed into the battery. The method we implemented is a slightly more sophisticated variant that allows the output level to increase above R when there the wind power level is significantly above R for some time.

4 Results

We present results for estimates of the minimum battery capacity required depending on a given ramp rate limit and using the data from a small 30kW turbine at the CSIRO laboratories in Newcastle. This data set has power measured every second and we have selected a period of a million seconds to test our methods on. We present results for six different methods: simulation of the three on-line heuristics described

in Section 3; solving the TWEOP for the whole data set; using the descent method; and by solving CminLP and ACminLP using CPLEX. For CminLP and ACminLP the complete data sets are far too large to solve so the numbers reported are the maximum over 5 smaller data subsets of 2000 points each, selected to pick out some of the most variable periods. In order to evaluate the effect of this sampling we also tested the descent method on the smaller data sets and report the battery requirement found in this way as Descent(5). Figure 2 shows the

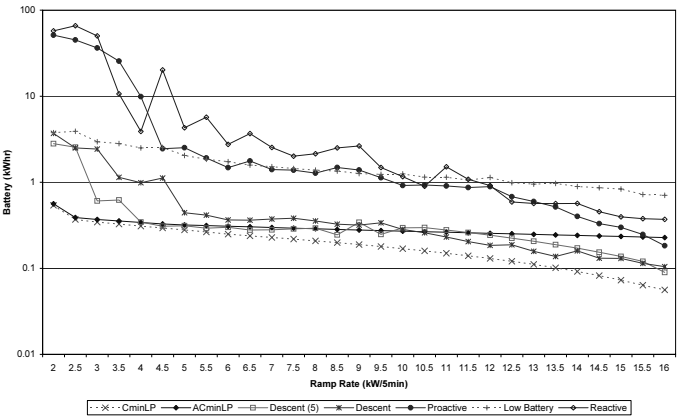


Fig. 2. Battery required (kWhr) for a 30kW as function of R (kW/5min)

battery required in order to avoid ramp rate violations. In principle all lines should be strictly deacreasing. However due to the heuristic nature of the (non-LP based) algorithms, sometimes more storage is required when the ramp rate limit is increased slightly. The maximum ramp rate allowed has a significant impact on the effectiveness of the various algorithms. For very small ramp rates (less than about 5kW/5min), the descent method produces quite poor solutions compared to the LP methods, some of which can be explained by the choice of samples but some of this difference is due to the inexact nature of the descent method. The on-line heuristics perform badly with the exception of the Low Battery heuristic.

In the middle range of ramp rates (5-12), the descent method is very similar in performance of the ACminLP method though slightly worse than the CminLP. The on-line heuristics have similar performance with the Proactive and Low Battery methods being slightly better than the Reactive method.

For high ramp rates the Low Battery method and ACminLP methods become quite ineffective, with the Proactive method dominating the on-line heuristics. The descent method on the whole data set produces some even lower results than the Descent(5) method, as the latter is constrained to start each sample with a half full battery.

In terms of computational efficiency a single run of the on-line heuristics (with a given battery capacity) requires about a second of CPU time on a 2GHz PC to process a million points, while the descent method takes about 10 times as long. Multiple runs with different battery capacities were used to determine the minimum battery requirements. In contrast the ACminLP method requires about a second of CPU time per thousand points, with the CminLP being significantly slower again.

5 Conclusions

We have presented an application arising from the current global trend towards increasing the participation of wind generators in the electricity supply mix. A mathematical formulation and several methods are presented for determining battery capacity that can achieve given level of smoothness in wind energy. The results show that battery requirements grow exponentially as the maximum ramp rate decreases, though reasonable ramp rates can be maintained with less than 10% of the maximum hourly output. The computational results also indicate that there is no single "best" approach that is going to work equally well across all data sets and operating conditions. Further research is still needed for example to determine whether forecasts can be used to improve the performance of the on-line algorithms[1]. More work also needs to be done to make the mathematical models more realistic and to incorporate constraints of the energy storage technology into the formulation.

References

1. Kent A, Mercer D (2006) Australia's mandatory renewable energy target (MRET): an assessment, Energy Policy 34(9):1046–1062
2. Kaiser R (2007) Optimized battery-management system to improve storage lifetime in renewable energy systems, Journal of Power Sources 168(1):58–65

[1] Early work by the authors indicate that wind energy output is difficult to predict and that poor forecasts can reduce the effectiveness of algorithms.

The Influence of Social Values in Cooperation

Robert Feyer[1], Ulrike Leopold-Wildburger[2], and Stefan Pickl[3]

[1] SIEMENS Vienna. robert.feyer@fin4cast.com
[2] University of Graz. ulrike.leopold@uni-graz.at
[3] University of the Federal Armed Forces Munich. stefan.pickl@unibw.de

Summary. This paper deals with the topic of cooperation among economic agents in a repeated game with unknown length. There is still little empirical evidence why and under which conditions people cooperate at all in situations such as the prisoner's dilemma game (PDG). There is extensive literature on the theory of infinitely repeated games and also a large number of explanations for cooperative behaviour. Experimental evidence on the question how future features will affect the behaviour is rare and the classification of the participants into social and selfish oriented types is new in this context.

Modern contract theory deals a lot with the willingness to cooperate. Therefore we raise the question under which conditions cooperation is created in special economic situations (for example in energy contracts).

The present study examines the influence of pre-existing individual differences in social value orientations measured by the outcomes to oneself and others according to the ring measure by McClintock [10]. We run an experiment in the lab and we are able to figure out the high percentage at cooperation in a PDG in which the number of future rounds is unknown and the fact that cooperation is significantly dependent on the type of the subjects' social value orientation.

1 Introduction

In situations in which there is always a future as it can be simulated by repeated games with unkown length the credible threat of future retaliation can cause opportunistic behaviour and it can be a reason for supporting cooperation. Dal Bo (2005) [1] calls it 'the shadow of the future'. In our paper, we report on a series of an experiment which consists of 2 parts: a questionnaire which aims to classify the participants into categories used by psychologists and the repeated prisoner's dilemma (PD) game by which the cooperative and defective behaviour

respectively can be measured. It will be shown that according to Dal Bo [1] the possibility of future action modifies the players' behaviour; however we figure out that this result is significantly depending on the classification of the player. The pro-social value orientation of a player realizes far fewer opportunistic actions and supports significantly more often cooperation than the selfish type does.

2 Social Value Orientation

It has been established that there are individual differences in preferences for expressing social values (e.g., [4]), that social values systematically affect choice behaviour in two person situations (e.g., [5]) and in n-person experimental games (e.g., [7, 6]).

Kelley and Thibaut [3] as well as Olekalns and Smith [11] provide insight into the social values that underlie the social decision making of individuals in outcome interdependent situations. Social values can be defined as distinct sets of motivational or strategic preferences among various distributions of outcomes for self and others [8]. Traditionally the following social values derived from the conceptual framework formulated by Griesinger and Livingston [2] and McClintock [9] are distinguished: altruism, cooperation, individualism, and competition.

According to Kuhlman and Marshello [5] and Liebrand and Van Run [7] four functions have been identified as utility functions that underlie the choice behaviour of a significant proportion of subjects in interdependent decision tasks.

Social values are defined as distinct sets of motivational or strategic preferences with the weighting rule [8] depending on the weights w_1 and w_2:

a) altruism is maximizing another's outcome ($w_1 = 0$, $w_2 = 1$);
b) cooperation, maximizing joint outcomes ($w_1 = 1$, $w_2 = 1$);
c) individualism, maximizing one's own outcome ($w_1 = 1$, $w_2 = 0$) ;
d) competition, maximizing one's outcome relative to other ($w_1 = 1$, $w_2 = -1$).

To investigate differences in the interpretation of others' behaviour in interdependency situations, a subject's choice behaviour is separated into two components, the outcomes the subject chooses for oneself and the outcomes that are chosen for the other. This allows us to see if the socialisation effect obtains for both the outcomes given to self and the outcomes given to the other: pro-social and pro-selfish.

3 Experimental Design and Procedure

We conducted a prisoner's dilemma experiment with the payoffs given in Table 1 in which players interacted repeatedly with the same partner. The continuation rule, however, is unknown to the participants. After a certain number of rounds not known to the participants in advance, each player was matched with a new partner (s)he did not play against before. Four players participated in a session. Therefore each player was matched with a different opponent three times, i.e., we had 3 matchings per session. Each matching consisted of 15 repeated interactions, whereas the number of interactions has not been known by the participants.

The experimental protocol we used in all sessions was as follows: Each subject had to make a decision. (S)he repeatedly played against another subject in a dyad. Inter-subjects contacts were anonymous, and the choices of the opponent group were shown on the screen. Each subject was paid the amount retained by his/her decision.

A well-known characteristic of the prisoner's dilemma game is the ambiguous relation between cooperation and performance. On the one hand, each player can increase her/his payoff by defecting instead of cooperating. On the other hand, the payoffs of all interacting players increase if a higher share of them cooperates. The theoretical basis for these contradicting tendencies is clearly established. In this study, we will analyse the empirical implications and characteristics of these ambiguous motives depending on the social values orientation of the subjects.

Table 1. Notation for outcomes and payoffs in the prisoner's dilemma game experiment. Each point (payoff) was converted into Euro at an exchange rate of 35 Eurocents

Row-player/column-player	Cooperation (A)		Defection (B)	
Cooperation (A)	c/c	(3/3)	s/e	(0/5)
Defection (B)	e/s	(5/0)	d/d	(1/1)

4 Hypotheses and Results

H1a: There is a positive correlation between cooperation and payoff.
H1b: Pro-social oriented participants attain a higher share of cooperation and consequently achieve higher payoff than pro-selfish oriented participants.

H2: If two pro-social oriented participants play against each other, they obtain a higher share of cooperation and payoff than two pro-selfish oriented participants or one pro-social and one pro-selfish oriented participant interacting / playing against each other.

The prisoner's dilemma game has been performed with 64 participants during the winter term 2005/06 at the University of Graz. We can summarize our findings in the following way: First, the questionnaire with the 24 questions requiring the participants to decide on one of two possible options referring to their social value orientation divided the participants into 2 groups: 32 participants were categorized as participants with pro-social value orientation and 32 participants with pro-selfish value orientation.

Second, the prisoner's dilemma game showed an obvious decline in cooperation within a matching indicated the attempt to exploit the opponent. There is obviously a decrease in the level of cooperation, especially within the second and the third matching. This is because of the so called end-effect. Each player tends to defect towards the end of the matching to exploit the opponent or to avoid being exploited.

4.1 Results Referring to H1a and H1b Payoff Depending on Share of Cooperation

There is a positive correlation between share of cooperation and payoff. Pro-social oriented participants attain a higher share of cooperation and consequently achieve higher payoff than pro-selfish oriented participants (see Table 2).

All values are statistically significant on the 0.1% level according to the Wilcoxon rank-sum test.

4.2 Results Referring to H2

If two pro-social oriented participants (forming a Matching Group 1) play against each other, they obtain a higher share of cooperation and payoff than two pro-selfish oriented participants (forming a Matching

Table 2. Average payoff and share in cooperation among both social value orientations

Social Value	Pro-social	Pro-social	Pro-selfish	Pro-selfish
Orientation	Cooperation	Payoff	Cooperation	Payoff
Mean	58.47%	99.97	38.68%	88.75
Standard dev.	26.78%	22.84	22.75%	20.21
Minimum	2.22%	50	0.0%	45
Maximum	100%	132	88.89%	130

Group 3) or one pro-social and one pro-selfish oriented participant (Matching Group 2) interacting / playing against each other.

Table 3 gives the means and standard deviations in all matching groups. The values of Matching Group 1 (MG 1) compared with Matching Group 3 (MG 3) show a significant difference in amount of payoff and percentages of cooperation according to the Wilcoxon rank-sum test (p smaller than 0.1% level).

Table 3. Means and standard deviations of payoffs and percentages of cooperation for all matching groups

Matching group (MG)	Social value	Mean	Standard deviation	Mean	Standard deviation
		Payoff	Payoff	Cooperation	Cooperation
MG 1	Soc:Soc	36.72	10.53	67.20%	33.34%
MG 2a	Soc:Self	29.63	10.32	48.99%	32.61%
MG 2b	Self:Soc	32.22	11.22	43.33%	32.13%
MG 3	Self:Self	27.16	9.61	34.40%	28.96%

5 Summary

The results revealed that participants with pro-social orientation, contrary to those with pro-selfish orientation were perceived as more coop-

erative and were able to manage higher payoffs. The further effect appeared to be more pronounced among pro-selfs rather than pro-socials: The matching of two pro-selfish oriented participants leads the lowest percentage of cooperation and also significantly the lowest payoff.

References

1. P. Dal Bo. Cooperation under the shadow of the future: Experimental evidence from infinitely repeated games. *American Economic Review*, 95(5):1591–1604, December 2005. Available at http://ideas.repec.org/a/aea/aecrev/v95y2005i5p1591-1604.html.
2. D. W. Griesinger and J. W. Livingston. Toward a model of interpersonal motivation in experimental games. *Behavioral Science*, 18:173–188, 1973.
3. H. H. Kelley and J. Thibaut. *Interpersonal relations: A theory of interdependence*. Wiley, New York, 1978.
4. G. P. Knight and A. F. Dubro. Cooperative, competitive, and individualistic social values: An individualized regression and clustering approach. *Journal of Personality and Social Psychology*, 46(1):98–105, 1984. Available at http://cat.inist.fr/?aModele=afficheN&cpsidt=9697715.
5. D. M. Kuhlmann and A. F. J. Marshello. Individual differences in game motivation as moderators of preprogrammed strategy effects in prisoner's dilemma. *Journal of Personality and of Social Psychology*, 32:922–931, 1975.
6. W. Liebrand, H. Wilke, V. R., and W. F.J.M. Value orientation and conformity. A study using three types of social dilemma games. *Journal of Conflict Resolution*, 30(1):77–97, 1986.
7. W. G. B. Liebrand and G. van Run. The effects of social motives on behavior in social dilemmas in two cultures. *Journal of Experimental Social Psychology*, 21:86–102, 1985.
8. C. McClintock and E. van Avermaet. Social values and rules of fairness: A theoretical perspective. In V. J. Derlaga and J. Grzelaj, editors, *Cooperation and Helping Behavior. Theories and Research*:43–71. Academic Press, New York, 1982.
9. C. G. McClintock. Social motivation — a set of propositions. *Behavioral Science*, 17:438–454, 1972.
10. C. G. McClintock. Social values: Their definition, measurement and development. *Journal of Research and Development in Education*, 12:121–137, 1978.
11. M. Olekalns and P. L. Smith. Social value orientations and strategy choices in competitive negotiations. *Personality and Social Psychology Bulletin*, 25(6):657–668, 1999. Available at http://psp.sagepub.com/cgi/content/abstract/25/6/657.

Designing Sustainable Supply Chains by Integrating Logistical and Process Engineering Aspects – A Material Flow Based Approach for 2^{nd} Generation Synthetic Bio-Fuels

Grit Walther[1], Anne Schatka[1], Thomas S. Spengler[1], Katharina Bode[2], and Stephan Scholl[2]

[1] Department of Production & Logistics, Braunschweig University of Technology, Katharinenstr. 3, 38106 Braunschweig
{g.walther|a.schatka|t.spengler}@tu-bs.de

[2] Institute for Chemical and Thermal Process Engineering, Braunschweig University of Technology, Langer Kamp 7, 38106 Braunschweig
{ka.bode|s.scholl}@tu-bs.de

Summary. In this contribution, a concept for a techno-economical decision support system for the production and distribution of bio-fuels will be presented. The production of bio-fuels is carried out in multi-stage processes and thus, is characterized by complex production structures. Furthermore, a variety of economical and technical risks need to be considered during planning. Therefore, decision support methods considering logistical as well as process technology aspects in an integrated approach are required.

1 Introduction

By the year 2020, the EU directive 2003/30/EG requires a 20% substitution of fossil fuels through alternative fuels in the road traffic sector. This attempt is both ecologically and politically motivated.[1]

To attain this aim, the extensive build-up of efficient production capacities for bio-fuels will be necessary by the year 2020. Today, 1^{st} generation bio-fuels are being used. However, since their production is characterized by a low specific energy production per unit of biomass, they are expensive. In the long-term, a competitive alternative will be provided by 2^{nd} generation bio-fuels. The production of those is characterized by a high specific energy production, due to all cellulose input being converted into fuel. Furthermore, 2^{nd} generation bio-fuels

promise a good compatibility with today's and future engine generations. Today, production schemes for 2^{nd} generation bio-fuels are still under development or at a pilot plant stage. The current production schemes vary in processing steps, biomass treatment, gasification and synthesis processes. [2]

2 Network Planning for 2^{nd} Generation Bio-Fuels

Currently, potential investors and network planners are challenged by the configuration and future operation of production networks for 2^{nd} generation bio- fuels. Firstly, there is a lack of decision support methods that are able to consider the specifics of material conversion processes. Secondly, a range of uncertainties e.g. related to the new technologies, political decisions and market developments on the raw materials as well as on the product side need to be considered.

Adequate decision support methods need to consider the technical and logistical specifics of the potential sourcing, production and distribution processes in an integrated manner. Advanced-Planning-Systems (APS) provide economic decision models for an efficient planning and control of complex production and logistic networks [3]. However, up to now APS focuses mainly on planning tasks for the manufacturing industry, i.e. common production and assembly processes based on converging structures, which can be described sufficiently by bills of material and operation charts. However, since the production of 2^{nd} generation bio-fuels is carried out in material conversion processes, planning models for the processing industry are necessary. The processing industry is characterized by chemical transformations, converging, diverging as well as cyclic structures, joint production, recirculation and various intermediate products. Within single process entities no linear transformation functions exist that can be deduced based on reaction and phase equilibria. Furthermore, the possible substitutions between equipment related and energy related input factors lead to numerous input factor combinations that need to be considered. [4] In addition to these aspects, which are typical for processing industry, production of 2^{nd} generation bio-fuels is characterized by a variety of alternative options on the input, process and output side. Different feedstocks and feedstock qualities can be used, substitutional relations between process parameters exist, and the output relation of diesel and gasoline can be controlled within boundaries. The results are summarized in flow charts as fundamentals for the technical system design. Therefore, modeling can not be based on bills of material, but has to be based on material

and energy balances originating from the chemical and thermodynamic specifications of the single unit operations.

Uncertainties about new technologies are typically related to the long-term stability, selectivity and yield of the catalyst, leveling-up of trace components under recycling conditions, decreasing of equipment effectiveness due to fouling or corrosion ect.. These uncertainties may lead to alternative processing structures, equipment design and/or operating conditions, thus significantly affecting investments as well as operating costs. The input of biomass leads to uncertainties concerning the availability, consistency and price of biomass. These parameters depend not only on technical or availability aspects and on environmental effects, but also on developments and decisions in the political arena, and on developments in the food processing industry. Decisions about subsidies as well as long-term guidelines that go beyond the year 2020 have not yet been made. Finally, the competitiveness and thereby the distribution of 2^{nd} generation bio-fuels is directly related to the development of fossil fuel prices, which is very volatile.

Hence, the aim is to develop a techno-economical decision support system for the sourcing, production and distribution of 2^{nd} generation synthetic bio-fuels. This will take into account technical and logistical specifics of the supply chain for the production of 2^{nd} generation bio-fuels as well as related uncertainties.

3 Planning Concept

Against this background, a hierarchical planning concept for integrated technical and logistical system design is presented in the following. The concept is based on coupling material and energy balancing models and economic decision support models for long-, mid- and short-term planning problems [see Figure 1].

On the long-term planning level, production facility and capacity allocation in a staged and distributed production network are determined. For this purpose, a multi-stage dynamic warehouse location problem will be developed. Input data to this model are long-term market forecasts, distances between potential network nodes, existing structures, technically feasible production schemes, potential in- and outputs of the processes, as well as estimated production, transportation and storage costs. However, not all necessary data is available, since no reference data from large-scale production capacities for 2^{nd} generation bio-fuels yet exist. Thus, relevant information on material and energy flows as well as technical parameters and costs must be

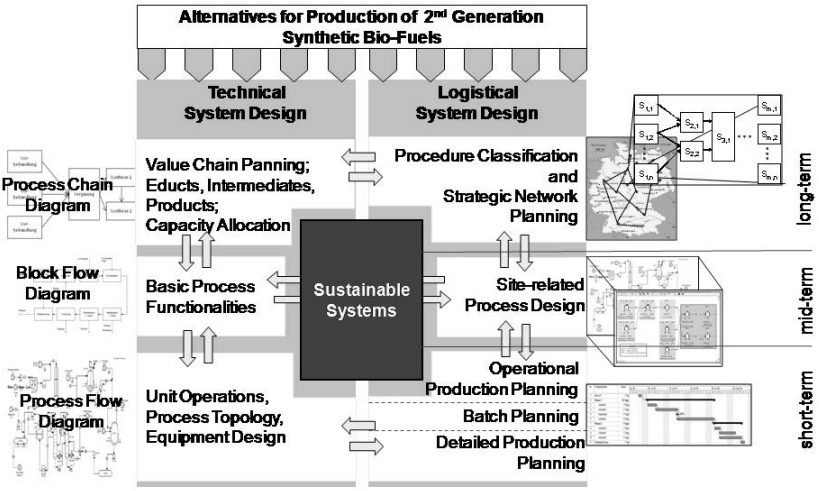


Fig. 1. Hierarchical planning concept for integrated technical and logistical system design

obtained from technical models, determining basic process operations, technical process parameters as well as interlinking and time coherences between logistical and conversion steps. In particular, options for the decoupling of processing steps and the results of the up-scaling of processes will be examined in order to provide information on decentralization options and economies of scale. Based on such technical models, simulations of material and energy flows will be carried out, and results will be integrated into strategic network planning models. If production processes can be decoupled, there are opportunities for decentralized first processing steps, e.g. pre-treating voluminous biomass such as straw and transporting only the resulting high-energy, low-volume intermediate goods. If technical models show that economies of scale can be realized, it might be beneficial to centralize processes for further processing steps.

The planned sites and capacities as nodes of a – decentralized or centralized – logistical network are the basis for the site-related plant configuration that is taking place at the mid-term planning level. From an economical point of view, decisions have to be taken whether a fixed or rather a flexible production structure should be implemented. Flexibil-

ity of the production structure may relate to the use of different educts or raw materials, optionally as intermediates after a pre-treatment, capacity variations accounting seasonal fluctuations, or shifts in product specification. The design options with different flexibility degrees have to be economically assessed through the modeling of specific production costs and returns for the alternatives. Therefore, more detailed information on material and energy flows, sourcing and distribution options, inter-stage distances, and reliability assessments of single plant components as well as mid-term market forecasts are needed. Technical flow sheeting models provide the material and energy flow information. [5, 6] In flow sheeting systems, the technical configuration of conversion and separation processes will be modelled based on unit operations like chemical reactions, rectifications, and extractions building on reaction, physical property and phase equilibrium informations. As a result local material and energy flow data, along with the corresponding processing conditions (pressure, temperature, and concentrations) will be given. With these detailed flow sheeting models, intra-process material and energy flow information becomes available, and restrictions on the operational mode are revealed, providing information on flexibility constraints of the processes, which are needed for economic evaluation.

On the short-term level, production planning and scheduling has to be carried out. In the literature, specific economic planning models for the processing industry have been developed, so far mainly for the scheduling of batches and charges. Within these models, batch and charge sizes are determined and the sequencing of available resources is carried out, taking into account demand as well as technical restrictions. Until now, only a few models have been developed for continuous and semi-continuous production [7], although the associated planning problems are of great importance for processing industry. Additionally, within existing models for scheduling in the processing industry, technical restrictions are only implicitly taken into account, as specific restrictions of the optimization problem. This leads to customized models, which need to be changed as soon as even minor changes in the plant structure or in process parameters (e.g. pressure, temperature or dwell time) occur. In reality, operational parameters can not be considered as static variables, but instead are subject to change with varying batch size and sequence. Thus, it is necessary to explicitly consider technical restrictions within models. Therefore, we aim to develop generic models by explicitly integrating technical aspects into production planning based on the integrated approach developed by [8]. Based on flow-sheeting models of the procedural processes, a

nonlinear planning model with a monetary objective function will be developed. The interrelation of inputs and outputs is described using thermodynamic, stochiometric, and technical transformation functions. Decision variables are the monetarily evaluated inputs and outputs, as well as technical model variables such as disintegration or integration of process steps or the allocation of capacities. The approach of [8] will be extended incorporating scheduling aspects.

4 Conclusions

The extensive build-up of production capacities for 2^{nd} generation bio-fuels due to politically and ecologically motivated guidelines presents decision makers with a range of challenges. In this enviroment, techno-economical decision support models for a sustainable network configuration and operation for the production of 2^{nd} generation bio-fuels are urgently needed. In this paper, a hierarchical framework was presented that provides the integration of technical and economical models. Within this framework, sophisticated planning models have to be developed and applied to the design and control of supply chains for 2^{nd} generation bio-fuels. This is part of an ongoing research project.

References

1. Richtlinie 2003/30/EG des europäischen Parlaments und Rates vom 8. Mai 2003 zur Förderung und der Verwendung von Biokraftstoffen im Verkehrssektor
2. Deutsche Energie-Agentur GmbH (2006) Biomass to Liquid - BtL Realisierungsstudie
3. Fleischmann B, Meyer H (2003) Planning Hierarchy, Modelling and Advanced Planning. In: de Kok AG ,Graves C (eds) Supply Chain Management: Design, Coordination and Operation. Elsevier, Amsterdam et al.
4. Dyckhoff H, Spengler TS (2007) Produktionswirtschaft. Springer, Berlin et al.
5. Spengler TS (1998) Industrielles Stoffstrommanagement. Erich Schmidt, Berlin
6. Rentz O (1979) Techno-Ökonomie betrieblicher Emissionsminderungsmaßnahmen. Erich Schmidt, Berlin
7. Hermann S, Schwindt C (2007) Planning and schedueling continuous in the process industries. In: Günther HO, Mattfeld DC, Stuhl L (eds) Management logistischer Netzwerke. Physika, Heidelberg
8. Penkuhn T (1997) Umweltorientiertes Stoffstrommanagement in der Prozessindustrie. Peter Lang, Wien et al.

Part IX

Entrepreneurship and Innovation

About the Limitations of Spreadsheet Applications in Business Venturing

Benjamin B. Gansel

PricewaterhouseCoopers Corporate Finance Beratung GmbH,
Marie-Curie-Straße 24-28, 60439 Frankfurt am Main, Germany
benjamin.gansel@de.pwc.com

1 The Mirage of Spreadsheet Applications

Spreadsheets are ubiquitously applied in end-user applications ranging from simple adding calculations to complex decision problems. As most people avoid utilizing different software packages for each problem, they routinely use (Excel) spreadsheets to perform modeling or to aid decision making. The wide acceptance of spreadsheets has also led to an overwhelming application in financial planning in business venturing. Almost all standard entrepreneurship textbooks (see for example, Barringer et al., 2005; Hisrich et al., 2005; Scarborough et al., 2006) propose (Excel) spreadsheets to create a financial plan as part of a business plan. The major advantage of spreadsheet applications is the ease of handling and the automation of calculation.

Beside some practical weaknesses the user faces the major and most obvious drawback by opening a spreadsheet that contains a financial plan of a start-up. The user instantaneously sees forecasted numbers without understanding the coherences behind these values. They are assigned to functions and one can see either the numbers or the functions, but not both (Denardo, 2001). To understand the business concept of these values, the user deeply digs into all worksheets of the spreadsheet model and the underlying interrelated functions, which is both difficult and time consuming. The user develops some kind of an imaginary graphical image of the financial plan. Thus, a spreadsheet model consisting of forecasted values conveys a mirage with respect to the actual business concept. It encourages users to focus on the forecasting numbers themselves by leaving out a fundamental understanding of the strategic interrelationship between complex financial and business decisions. Entrepreneurs need to understand such complex interrela-

tionships as entrepreneurial decisions are in most cases characterized by one-off new situations requiring subjective judgment under a limited amount of information. The primary aim of such planning approaches in entrepreneurship literature is the creation of pro forma financial statements.

The question arising is whether there is a superior approach that overcomes these drawbacks. At the same time the approach has to create a pro forma financial plan. Concerning the issues the paper proposes modeling with influence diagrams, particularly at the graphical level which goes beyond the narrow boundaries set by conventional planning with spreadsheets. Beside their representational compactness, influence diagrams are intuitive to understand and facilitate the formulation, assessment, and evaluation of decision problems as perceived by decision makers (Howard, 1990; Shachter, 1986). They describe the structure of a decision problem in a concise way, and are effective means for communicating with decision makers, computers, among people, and experts (Howard, 1990; Kirkwood, 1992; Owen, 1978). There exists an important distinction between the ubiquitous applied decision making tool 'spreadsheet' and the decision making methodology 'influence diagram'. The paper proposes the application of the methodology that may be implemented by spreadsheet tools.

2 Beyond Conventional Spreadsheet Applications: An Illustrative Example of an Influence Diagram Model

Conventional spreadsheet applications are used in several standard entrepreneurship textbooks. A general income statement is derived according to Barringer and Irland (2005), Hisrich et al. (2005), Scarborough and Zimmerer (2006). It contains numerical predictions based on deterministic functions. Such an income statement only allows manipulation of the values, but it cannot model the inherent financial and strategic interrelationships and decisions. It provides no methodology to aid decisions and makes only a negligible contribution to understand the business concept. The spreadsheet model operates solely at the numerical and functional level of specification. To gain a deeper insight into the decision problem the concept of the value of perfect and imperfect information is applied. Modeling a situation with perfect information provides an upper boundary for gathering real information. In addition, the influence diagram has to be transferred into Howard's canonical form (Howard, 1990; Howard et al., 1981). This allows evaluation of any information gathering as the modeling does not create

a loop. The value of perfect information can be determined for each chance node separately, in combination and in total. For the example, table 1 provides an overview of values of perfect information given different uncertain variables. The value of perfect information of market share given high price is more than twice as much as the value of market share given low price which provides important information for real information gathering. Perfect information of the variable market size is worthless to the entrepreneur as it has no impact on the optimal pricing strategy decision. This provides guidance of where to focus real information gathering.

Table 1. Values of perfect information

Variable	**Expected value**	**Value of perfect information**
Market share given high price	47,775	19,150
Market share given low price	36,690	8,065
Market size	28,625	0
Market share given high price and market share given low price	51,675	23,050
Market share given high price and market size	47,775	19,150
Market share given low price and market size	36,690	8,065
All variables	51,675	23,050

In real situations the information is usually not perfect. A market analysis for example results into imperfect yet relevant information. With such information the entrepreneur may be better able to estimate the potential market share. An influence diagram allows modeling of results of a market analysis. This requires assigning a conditional probability distribution on the analysis result given the variable of interest. A more convenient way allows the estimation of a general quality of the market analysis where the quality of such an analysis is measured by the probability that the analysis result is of perfect information. The probability p can take any value between zero and one, i.e. $0 \leq p \leq 1$.

A sensitivity analysis on the quality variable shows the value of imperfect information with respect to different qualitative levels of market analyses (shown in Figure 1). The diagram is distinguished into three regions indicating different optimal strategies. A market analysis with $p_I < 0.078$ (region I) has no value to the entrepreneur as it does not change the initial low pricing strategy. Region II is of value if p falls within $0.078 < p_{II} < 0.260$. This means that a market analysis with a relatively low quality level has still an impact on the optimal pric-

ing strategy. Region III consists of $p_{III} > 0.260$ where a high pricing strategy is optimal given favorable and neutral analyses results.

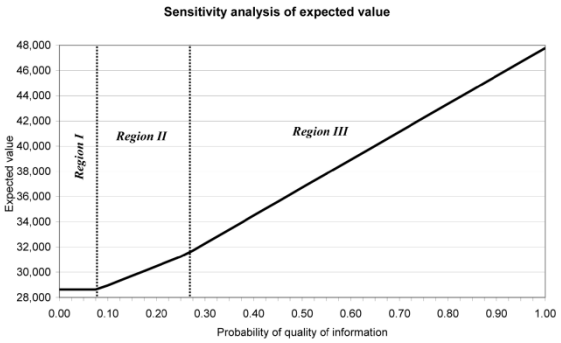


Fig. 1. Sensitivity analysis on the quality variable

3 Implications

The paper introduced a simplified pro forma income statement which does not provide a decision aid and operates solely at the numerical and functional level. A method that allows effective yet convenient modeling of a decision problem still requires the creation of an income statement. This can be extracted from an influence diagram. It is assumed that the entrepreneur has access to a market analysis given high price with an accuracy level of 0.5 incurring costs of \$5,800. As this is below the upper value of imperfect information of such a quality level the entrepreneur performs the analysis. A favorable and neutral analysis outcome results into a high pricing strategy whereas an unfavorable outcome results into the opposite strategy. The influence diagram model allows the creation of pro forma income statements for different decisions and scenarios.

4 Conclusion and Discussion

The paper argues that common spreadsheet models, proposed by standard entrepreneurship textbooks consisting of a financial plan for start-ups, encourage entrepreneurs to focus on the forecasting numbers themselves by leaving out a fundamental understanding of the strategic interrelationship between complex financial and business decisions inherent to a financial planning model. The primary aim is the creation of a pro forma financial statement. This is, however, a limited approach.

The paper proposes modeling with influence diagrams. The entrepreneur gains insight by means of a decision model about something that the entrepreneur was not aware at the beginning. An improved understanding results ultimately into better decisions. The concept of the value of perfect and imperfect information is applied to show that uncertain variables have different values of information. This provides guidance of where to focus real information gathering. The paper also introduces a quality parameter that allows modeling and comparison of any information gathering ranging from a perfect to a useless one. A pro forma income statement can still be extracted from an influence diagram model. Differently from conventional spreadsheet approaches, such an income statement represents the logical result of the decision maker's subjective and unique decision making process. This goes beyond the narrow boundaries set by conventional planning with pure spreadsheet models in business venturing.

References

1. Barringer, B. R. and Ireland, R. D. (2005), *Entrepreneurship successfully launching new ventures*, Pearson/Prentice Hall, Upper Saddle River, NJ.
2. Denardo, E. V. (2001), *The science of decision-making: A problem-based approach using Excel*, OR/MS Today, 28(4).
3. Hisrich, R. D., Peters, M. P. and Shepherd, D. A. (2005), *Entrepreneurship*, 6th ed., McGraw-Hill/ Irwin, Boston, Mass.
4. Howard, R. A. (1990), *From influence to relevance to knowledge* in Oliver, R. M. and Smith, J. Q. (Eds.), Influence diagrams, belief nets and decision analysis: Proceedings of the conference entitled "Influence Diagrams for Decision Analysis, Inference and Prediction", 3-23, Wiley, Chichester.
5. Howard, R. A. and Matheson, J. E. (1981), *Influence diagrams* in Howard, R. A. and Matheson, J. E. (Eds.), Readings on the principles and applications of decision analysis, 2:719-762.
6. Kirkwood, C. W. (1992), *An overview of methods for applied decision analysis*, Interfaces, 22(6):28-39.
7. Owen, D. L. (1978), *The use of influence diagrams in structuring complex decision problems* in Howard, R. A. and Matheson, J. E. (Eds.), Readings on the principles and applications of decision analysis, 2:763-771.
8. Scarborough, N. M. and Zimmerer, T. W. (2006), *Effective small business management: An entrepreneurial approach*, 8th ed., Pearson/Prentice Hall, Upper Saddle River, NJ.
9. Shachter, R. D. (1986), *Evaluating influence diagrams*, Operations Research, 34(6):71-883.

A Decision-Analytic Approach to Blue-Ocean Strategy Development

Matthias G. Raith, Thorsten Staak, and Helge M. Wilker

Department of Economics and Management, Otto-von-Guericke University, P. O. Box 4120, 39016 Magdeburg, Germany
raith@ovgu.de, thorsten.staak@ovgu.de, wilker@ovgu.de

1 Introduction

The potential value created with a new product or service provided by a firm is given by the difference between its (monetary) benefit, in the view of the firm's customers, and the unit production cost to the firm. To what extent this potential value can be exploited as a market opportunity depends on the firm's success in obtaining a competitive advantage over other firms in the market. In order to acquire a competitive advantage, a firm must outperform its rivals in value creation (cf. Besanko et al. [1]).

In order to enhance value creation, the firm has two generic options: It can raise the value for the customer, e. g., by differentiation of the product's features, or it can lower the costs for providing the product. However, as Porter ([5], p. 18) points out, 'achieving cost leadership and differentiation are (...) usually inconsistent, because differentiation is usually costly.' Therefore, the firm should view the two generic strategies as alternatives between which it must make a choice, since otherwise it may become 'stuck in the middle,' thereby sacrificing its competitive advantage.

It appears somewhat surprising, though, that Porter's view on the trade-off between differentiation and cost leadership is taken for granted in most of the literature on business strategy. From a decision-analytic perspective, it fails to acknowledge the multiple dimensions which typically characterize the benefits of consuming and the costs of producing a product or service. Hence, it should seem quite natural to consider a firm differentiating its product in one aspect while simultaneously reducing costs in another. Strategy selection then becomes a multi-attribute decision problem with important implications for mar-

ket analysis. Since the weights that consumers attach to the different attributes of a product depend on the consumer group that makes up the market, value creation can be influenced by shifting the market focus to other (potential) customers that not only place different weights on the attributes, but also value different attributes.

Kim and Mauborgne [2] offer an innovative approach to strategy development along this line. At the core of their approach is the realization that it is much more valuable for firms to focus their energy on finding or creating new, uncontested market space (a 'blue ocean') than to compete against incumbent firms on existing markets ('blood-red oceans'). The main instrument for finding blue oceans is a 'strategy canvas,' a visual depiction of strategies as value curves allowing the comparison and differentiation of industries and competitors.

However, the strategy canvas, as presented by [4], seems useful mainly for ex-post diagnosis and explanation of successful blue-ocean strategies. For the strategy developer in the ex-ante perspective it remains unclear how exactly a blue-ocean strategy is recognized among possible alternatives. This is particularly a problem in an entrepreneurial context, where an unencumbered start-up usually has a large space of strategy choices. In order to use the qualitative concept of the strategy canvas for strategic planning, we consider the selection of a strategy profile as a multi-attribute decision problem. Within this framework the blue-ocean strategy can then be derived as the optimal choice. The decision-analytic approach allows the entrepreneur to quantitatively derive the optimal market strategy from the preceding market analysis. Moreover, sensitivity analysis allows one to test the robustness of the strategy with respect to changes in the relevance of strategy factors and their values. In addition, the quantification of the strategy canvas enables one to assess its impact on the market potential of different customer groups.

2 The Strategy Canvas: A Qualitative Tool for Ex-post Strategy Diagnosis

The *strategy canvas*, as developed by Kim and Mauborgne [4], depicts strategies as *value curves* allowing the comparison and differentiation of industries and competitors. As an example, Fig. 1 shows the strategy canvas for the U. S. wine industry in the late 1990s, and illustrates the strategy followed by Australian wine maker Casella Wines in entering the U. S. beverage market.

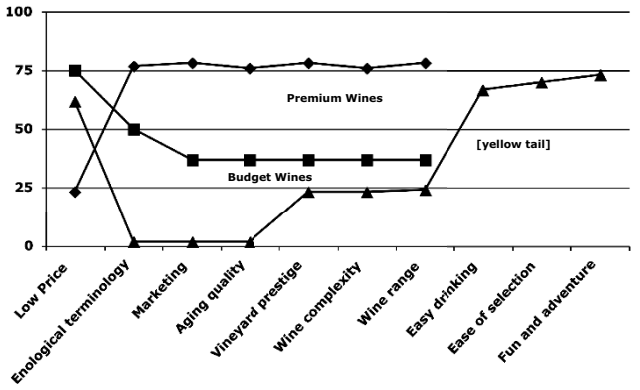


Fig. 1. U.S. wine industry in the late 1990s (cf. [4], 32)

The strategic factors that the industry competes on and invests in are displayed on the horizontal axis in Fig. 1. The vertical axis shows the offering level buyers receive for each strategy factor. Thus, strategic factors can be viewed as attributes of the products or services being offered. Competitors are judged on the level of their offerings to the customer in each factor and may be sorted into strategic groups, which are defined by equal or similar offering levels in each strategy factor. The result is a set of value curves for competing strategic groups in a given industry.

Fig. 1 shows two major strategic groups within the US wine industry. One group of firms competes on premium wines, investing heavily in strategic factors that allow them to differentiate their products. The other group offers budget wines, competing with low-value products that are also cheaper. In order to obtain a competitive advantage in this industry, a firm would have to offer a product with a higher value curve at overall lower unit production costs. The more competitive (blood red) the market is, the more difficult this is to achieve.

Kim and Mauborgne [2], [4], therefore, suggest to shift the market focus to alternative customers, who place less weight on the competitive strategy factors under consideration, while valuing other (new) strategy factors more strongly. As a consequence, investments can be reduced or eliminated altogether for strategy factors of lower importance, while relevant strategy factors can be expanded and new ones created. In Fig. 1 the multi-dimensional strategy profile (value curve) for Casella Wine's new product [yellow tail] illustrates the contrast to the two dominating strategic groups. By shifting the customer focus to non-wine drinkers, more interested in 'easy drinking,' 'ease of se-

lection' as well as 'fun and adventure' than traditional wine drinkers, Casella Wines was able to reduce costs in less valued strategic factors, which were of high relevance for the traditional wine market. The multi-dimensional strategy profile, thus, allows strategy developers to achieve both cost reductions as well as product differentiation at the same time. This is the essence of value innovation (cf. [3]), driven by consumer preferences rather than technology.

3 The Strategy Canvas: A Quantitative Tool for Ex-ante Strategy Development

The strategy canvas, as conceived by Kim and Mauborgne [4] is an elegant qualitative and didactic tool for understanding and explaining, ex post, the strategic deviation of high performers from traditional market incumbents in creating value. However, ex-post it is easy to argue that a firm's deviating strategy must have been 'better' than those of its competitors, knowing that the firm turned out to be successful.

In contrast, a strategy developer, e. g. an entrepreneur analyzing his or her entry into the market, ex ante, usually has a range of strategy alternatives for the new venture. The task in this case is to find the optimal strategy, given the observed strategies of competitors. If strategy curves are viewed as bundles of offerings for relevant strategy factors, the optimal strategy becomes the solution to a multi-attribute decision problem. We, therefore, propose to interpret the strategy canvas as a quantitative assessment of strategy profiles.

In Fig. 1 we interpret the different strategy factors as attributes of the offered products, which are valued and also weighted differently by customers. The vertical axis would then measure the benefit of the attribute, as perceived by customers. In addition, the individual attributes of the product, i. e. the strategy factors for the firm, must be weighted according to their importance to the customer, in order to assess the overall value of a strategy profile (for the customer) and, thus, to allow meaningful comparisons between strategy alternatives. By using standard multi-attribute rating methods or conjoint analysis, strategy alternatives (represented by value curves in the strategy canvas) can be quantified and ranked according to their overall values.

Both the valuation and the weighting of strategy factors from the customers' perspective are subjective. However, by interpreting the strategy canvas as a multi-attribute characterization of the firm's strategy alternatives, one can employ sensitivity analyses in order to test

the robustness of the firm's decision, i. e. its selection of a strategy alternative.

In competing in value creation, one can see how the firm benefits from shifting its market focus. As new customers place positive value on new attributes, previous attributes decrease in importance. In Fig. 1 the value curves of competitors remain the same, but the strategic factors they are focused on receive less weight. As a consequence, the existing strategies of competitors decline in overall value. As market competitors lose their bite, the ocean in which the strategy-developing firm is swimming becomes more blue.

4 Conclusion

The multi-attribute nature of the strategy canvas overcomes the one-dimensional, either-or decision over generic strategies, thus providing a more differentiated perspective of value creation. Value innovation is induced by adding new strategy factors which enable the innovator to deliver higher value than existing competitors, while offsetting costs by eliminating other factors which are not valued by customers. Yet, the strategy canvas in its basic form is a descriptive tool useful mainly for an ex-post characterization of successful market strategies.

By expanding the qualitative concept of the strategy canvas for the quantitative measurement of consumers' preferences, we showed how the strategy developer can perform ex-ante comparisons of strategy alternatives to select an optimal strategy. Uncertainties introduced by the subjective valuation and scoring methods can be minimized by using sensitivity analysis to test the robustness of the final selection.

References

1. David Besanko, David Dranove, Mark Shanley, and Scott Schaefer. *Economics of strategy*. J. Wiley & Sons, Hoboken, NJ, 4th edition, 2007.
2. W. Chan Kim and Renée Mauborgne. Blue ocean strategy. *Harvard Business Review*, 82(10):76–84, October 2004.
3. W. Chan Kim and Renée Mauborgne. Value innovation. the strategic logic of high growth. *Harvard Business Review*, 82(7/8):172–180, July–August 2004.
4. W. Chan Kim and Renée Mauborgne. *Blue ocean strategy : how to create uncontested market space and make the competition irrelevant*. Harvard Business School Press, Boston, Mass., 2005.
5. Michael E. Porter. *Competitive advantage : creating and sustaining superior performance*. Free Press, New York, 1985.

Flexible Planning in an Incomplete Market

Peter Reichling, Thomas Spengler, and Bodo Vogt

Otto-von-Guericke-University Magdeburg, Postfach 4120, 39016 Magdeburg, Germany. Peter.Reichling@ovgu.de, Thomas.Spengler@ovgu.de, Bodo.Vogt@ovgu.de

1 Introduction

In many cases, the financial situation of start-up companies only allows equity financing of growth investments. Every increase in capital needs a post-money valuation of the firm. Therefore, it is important to have appropriate methods to evaluate these companies. Firm valuation methods include flexible planning techniques and real option approaches. These methods are based on the present value method and/or the binomial model. They assume that the firm's cash flows can be duplicated by financial contracts and, therefore, imply a complete market, at least for this market segment.

However, due to choices of action by the management, start-up companies often show cash flow streams that cannot be duplicated at the financial market. From an option pricing point of view, this incompleteness can be interpreted as an ambiguous martingale measure to value the firm's state-dependent payoffs. On the other hand this ambiguity can be regarded as a vague outcome. Hence, an exact valuation of the company is not possible. Our model allows to determine bounds of the firm's value, if only an incomplete market exists, by taking option-style alternatives of the management into consideration.

2 Rigid and Flexible Planning

To illustrate our approach, we proceed with an example and assume an entrepreneur in the IT industry who plans to create a software. The cash flow of his project depends on the future economic situation and his decisions. In a simple two-period setting, the entrepreneur faces the sequential decision problem shown in Figure 1.

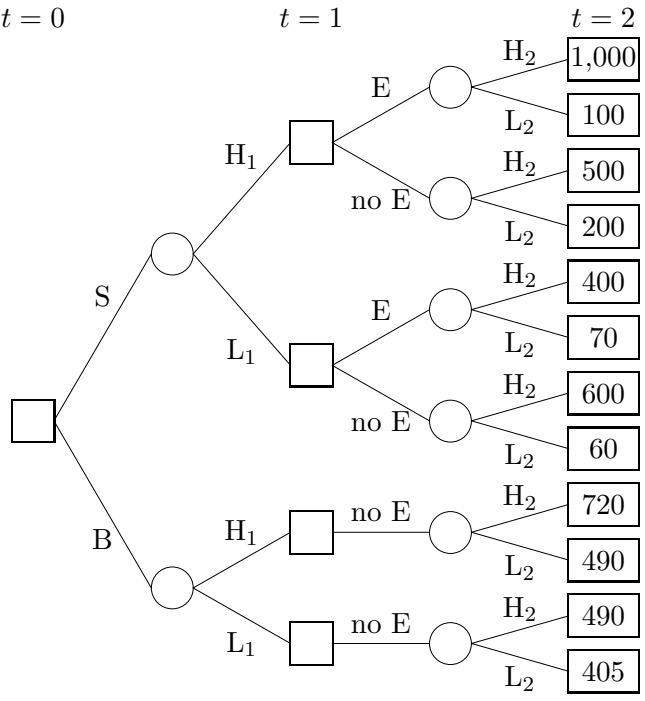


Fig. 1. Decision tree

At the beginning at point in time $t = 0$, the entrepreneur decides between action S (creation of a small software) and action B (creation of a big software). Subsequently, a chance move decides whether the demand is high or low. High demand H_1 occurs with physical probability $p(H_1) = 0.8$ and low demand L_1 arises with probability $p(L_1) = 0.2$. In $t = 1$ the entrepreneur decides on extending (E) or not extending (no E) the software, as shown in Figure 1. Afterwards, a second chance move determines the demand, where high and low demand H_2 and L_2 again occur with physical probability 0.8 and 0.2, respectively. This finally results in payoffs at $t = 2$.

According to Figure 1, the entrepreneur can choose one of the strategies $A_i = (a_0, a_1)$ with action choices a_0 and a_1 at $t = 0$ and $t = 1$, respectively: A_1 = (S,E), A_2 = (S,no E), A_3 = (B,no E), A_4 = (S,E if H_1 and no E if L_1), and A_5 = (S,E if L_1 and no E if H_1). The strategies A_1, A_2, and A_3 are rigid, whereas A_4 and A_5 represent flexible strategies. The corresponding decision matrix is shown in Table 1.

To value the strategies based on market prices we start with a complete financial market that consists of a risky and a risk-free asset. The

Table 1. Flexible planning with physical probabilities

Demand	H_1, H_2	H_1, L_2	L_1, H_2	L_1, L_2		
Prob p	0.64	0.16	0.16	0.04	Expected	
Strategy	Payoff				value	Rank
A_1	1,000	100	400	70	722.80	2
A_2	500	200	600	60	450.40	4
A_3	720	490	490	405	633.80	3
A_4	1,000	100	600	60	754.40	1
A_5	400	70	500	200	355.20	5

price of the risky asset in a binomial setting can only increase (up-state) or decrease (down-state) in every period by fixed factors. From risk-neutral valuation technique it is well-known that in this situation it is possible to compute risk-neutral probabilities $\hat{p}$ and $1 - \hat{p}$ for up- and down-state, respectively, as shown in the example in Figure 2. These risk-neutral probabilities avoid arbitrage opportunities between flexible planning projects and financial assets.

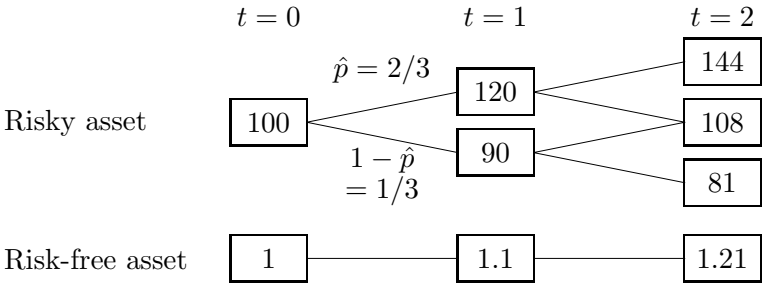


Fig. 2. Binomial market model

It is possible to compute the value of the flexible planning option based on market prices of financial assets (see, e.g., Dixit/Pindyck (1994) and Trigeorgis (1996)). Applying the risk-neutral instead of the physical probabilities to the project payoffs yields the results presented in Table 2. However, the expected values and the ranking of the strategies in Table 2 differ from those in Table 1. Employing option pricing theory, every strategy can be duplicated by a portfolio consisting of the risky and the risk-free asset. Therefore, the present values in Table 2 not only characterize fair values, but are also obtainable by replication via the financial market.

Table 2. Flexible planning with market-oriented valuation

Demand	H_1, H_2	H_1, L_2	L_1, H_2	L_1, L_2			
Prob $\hat{p}$	4/9	2/9	2/9	1/9	Expected	Present	
Strategy	Payoff				value	value	Rank
A_1	1,000	100	400	70	563.33	465.56	3
A_2	500	200	600	60	406.67	336.09	4
A_3	720	490	490	405	582.78	481.63	2
A_4	1,000	100	600	60	606.67	501.38	1
A_5	400	70	500	200	326.67	269.97	5

3 Valuation in an Incomplete Market

Below, we assume an entrepreneur whose payoff does not only depend on his decisions and on chance moves that determine the financial market development, but also on chance moves that merely affect his project. Imagine, e.g., that marketing activities determine whether the demand for his product increases with probability q, but these activities do not have any influence on the financial market. This situation might be regarded as realistic since most investors are too small to have an impact on prices at the financial market.

We model this situation by modifying the final payoffs in Figure 1. At least for some states, the final payoffs are binary lotteries $[q, \max, 1-q, \min]$ with $\max > \min$ instead of secure payoffs. The market prices of financial assets do not vary if the payoff of the project changes from max to min. In this situation, only an ambiguous market value of the project exists. We can derive market values if the final outcome is equal either to max or min. But due to its incompleteness, the financial market does not provide risk-neutral probabilities for these states so that we cannot combine both payoffs to receive an unambiguous market price. Instead, we only have a market-oriented value for the lower outcome min and the higher outcome max.

One possibility to deal with this ambiguity is to apply a preference based approach to the market-oriented values $V(\max)$ and $V(\min)$. One approach would be to assume a utility function $u(x)$ and compute the expected utility $\mathrm{E}(u([q, V(\max), 1-q, V(\min)]))$. A special case would be to assume a linear utility function which would result in a value of $q \cdot V(\max) + (1-q) \cdot V(\min)$. Thereby, we not only assume a linear utility function, but also have to use the physical probability q.

We also propose the Hurwicz (1951) criterion which was designed for the case of vagueness (see also Lindstaedt (2004)). This criterion

characterizes a person by a parameter α which describes his attitude towards ambiguity. The valuation resulting from this principle is $\alpha \cdot V(\max) + (1-\alpha) \cdot V(\min)$. The case $\alpha = 1$ corresponds to the maximax criterion, i.e. selecting the alternative that gives the highest payoff in the best cases. The case $\alpha = 0$ represents the maximin criterion.

If we look at our example in Figure 1 and only change the payoff for the small software strategy with extension in state H_1,H_2 from 1,000 to the binary lottery $[q, 1{,}000, 1-q, 800]$, we obtain the decision problem illustrated in Figure 3.

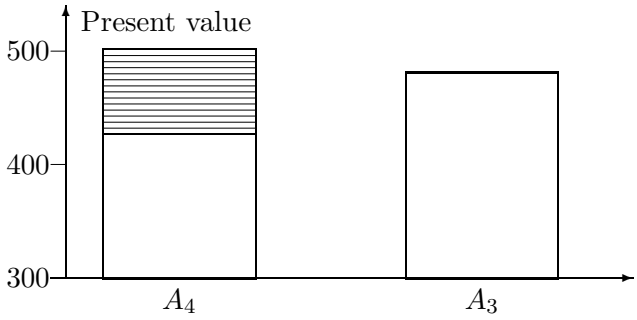


Fig. 3. Present value interval for vague payoffs

The value of strategy A_3 ("big software") is constant since no ambiguity is connected with this alternative. The value of strategy A_4 ("small software plus flexible extension") is vague. If the lower outcome of 800 occurs, A_3 is better and if the higher outcome of 1,000 arises, A_4 is superior. Obviously, the choice of the decision maker now depends on his preferences and/or his attitude towards ambiguity. However, the preference-based calculus should not be applied directly to the payoffs, but only to the ambiguous market-oriented values.

References

1. Dixit AK, Pindyck RS (1994) Investment under uncertainty. Princeton
2. Hurwicz L (1951) Optimality criteria for decision making under ignorance. Cowles Comission Discussion Paper, Statistics 370
3. Lindstaedt H. (2004) Entscheidungskalkuele jenseits des subjektiven Erwartungsnutzens. ZfbF 56:495–519
4. Trigeorgis L (1996) Real Options: Managerial Flexibility and Strategy in Resource Allocation. Cambridge (Mass)

Social Entrepreneurs, Lead Donors and the Optimal Level of Fundraising

Christoph Starke

Faculty of Economics and Management, Otto-von-Guericke-University Magdeburg, P. O. Box 4120, 39016 Magdeburg, Germany
christoph.starke@ovgu.de

1 Introduction

The relationship between donors and social entrepreneurs is often accompanied by diverse interest conflicts. According to Andreoni (1998, 2006), the significant and often largest part of the initial financial need of a social entrepreneur or a major new initiative of an existing charity originates from a lead donor. In this paper we consider a contracting situation in which the lead donor and the entrepreneur disagree on the optimal level of fundraising. More specifically, the lead donor dislikes high fundraising. Irrespective of the exact cause of the preference mismatch, the analysis examines how the grantor's direct regulation of the fundraising share affects the entrepreneurial decision calculus.

As a reaction to the restricted use of donations for fundraising, we find that the entrepreneur takes measures to become independent of donor claims by substituting for the leadership gift. In its extreme, fundraising remains unchanged, the size of the social project increases but the level of mission fulfilment is significantly lowered.

2 The Model

The social entrepreneur and the lead donor draw positive utility from the overall public good expenditures G. Without explicitly characterizing the decision calculus of the lead donor, her chosen size of the seed grant X enters the model as an exogenous parameter. The entrepreneur then decides on its allocation to the direct provision of the public good and further fundraising campaigns. Depending on the level of the fundraising investment F, the total further donations resulting

from these campaigns are denoted by $\psi(F)$, with $\psi(0) = 0$, $\frac{\partial \psi}{\partial F} > 0$ and $\frac{\partial^2 \psi}{\partial F^2} < 0$.

In addition to the previously characterized decision, the entrepreneur has the opportunity to charge a fee from service recipients, which generates the aggregated return S. Since such a fee would exclude the poorest from participation, her utility drawn from the provision of a given size of the public good is lower than without charging a service fee. This assumption is plausible since social entrepreneurs typically engage in problems in which a certain population group is assumed to be disadvantaged by the market allocation. By charging a fee, the social project approaches the market solution and, hence, is less desirable for the entrepreneur. The general utility function of the entrepreneur is given by

$$U(F) = \beta(S)G(F) \quad \text{with} \quad G(F) = X + S - F + \psi(F), \tag{1}$$

where β is a discount factor, with $\beta(0) = 1$ and $\frac{\partial \beta}{\partial S} < 0$.

We consider two versions of public good provision. In the first version, the entrepreneur covers expenditures exclusively by donations. Hence, with $S_1 = 0$ and $\beta(0) = 1$, the utility function reduces to

$$U_1(F_1) = G_1(F_1) = X - F_1 + \psi(F_1). \tag{2}$$

The second version features a positive service fee income $S_2 > 0$. With $\beta(S_2) < 1$, the utility function can be written as

$$U_2(F_2) = \beta(S_2)[X + S_2 - F_2 + \psi(F_2)]. \tag{3}$$

By taking into account that the entrepreneur can maximally spend the leadership donation as well as the service charges on fundraising, we can formalize her constrained optimization problem for each project version $i, i = 1, 2$:

$$\begin{aligned} \max_{F_i} \ & U_i(F_i) \\ \text{s.t. } & F_i \leq X + S_i \\ & F_i \geq 0 \end{aligned} \tag{4}$$

Without loss of generality, we assume that there exists an interior solution for the first and, hence, the second problem. Consequently, the optimal choices F_i^* satisfy the first order condition

$$1 = \frac{\partial \psi(F_i)}{\partial F_i}. \tag{5}$$

Because marginal costs and returns from fundraising are equal for both versions of public good provision, the optimal choices of solicitation expenditures coincide: $F_1^* = F_2^*$. Employing these quantities in equations (2) and (3) yields the condition by which the social entrepreneur prefers version one over two:

$$G_1(F_1^*) \geq \frac{S_2}{\frac{1}{\beta(S_2)} - 1}. \tag{6}$$

As a consequence, $U_1(F_1^*) \geq U_2(F_2^*)$ holds if the service fee revenues and the utility function parameter are relatively small. That is, the lower utility of one Euro public good expenditure in the second variant cannot be compensated by the additional income to the amount of S_2. This decision outcome is now taken to be the status quo of the following analysis.

3 Donor Restrictions on Fundraising Expenditures

Suppose now, the lead donor finds the chosen fundraising level of the social entrepreneur excessive and limits that part of the seed grant which can be used for solicitations. As Thornton (2006) observes, "[...] many government funders place explicit restrictions on the use of public funds by nonprofits, and these restrictions frequently prohibit nonservice spending, such as fundraising."

The entrepreneur is now permitted to use only $\overline{F}$ of the leadership gift to solicit for further donations. Certainly, an effective cap would require $\overline{F}$ to be strictly lower than F_i^*. Without services fees, this restriction prevents the entrepreneur from generating the maximum fundraising income and leaves condition (5) unmet. The scope of the public good provision and, equivalently, the entrepreneur's utility reduces to

$$U_1(\overline{F}) = G_1(\overline{F}) = X - \overline{F} + \psi(\overline{F}). \tag{7}$$

This inferiority is mitigated in the second version by the allocation of service fee income to fundraising until, given the charges are sufficiently high ($S_2 \geq F_i^* - \overline{F}$), the optimal level is reached again.

$$\begin{aligned} U_2(\overline{F}) &= \beta(S_2)[X + S_2 - \tilde{F}_2 + \psi(\tilde{F}_2)], \\ \text{with} \quad \tilde{F}_2 &= \overline{F} + S_2 - R^* \quad \text{and} \quad R^* \geq 0. \end{aligned} \tag{8}$$

The residual fraction, denoted by R^*, is directly invested into public good provision. Equation (8) elucidates that the entrepreneur also

draws less utility from the second project variant if $S_2 < F_i^* - \overline{F}$. However, compared to project version one, the absolute amount of reduction differs. Furthermore, from the comparison of utility equations (7) and (8) one can infer that public good provision without fees is still preferred if and only if

$$G_1(\overline{F}) \geq \frac{R^* + [\psi(\overline{F} + S_2 - R^*) - \psi(\overline{F})]}{\frac{1}{\beta(S_2)} - 1}. \tag{9}$$

The numerator on the right hand side characterizes the difference in the size of the public good between both project versions. In their maxima, the size of version two exceeds that of the first by the amount of the service fee income which is not used for fundraising, R^*, and the part of the fundraising cash flow which is generated by the invested service fee income $[\psi(\overline{F} + S_2 - R^*) - \psi(\overline{F})]$. A switch in the preference relation is now advanced by two factors. First, in the status quo the difference in the size of the public good between version two and one has been S_2. With the donor's restriction, the entrepreneur devotes the amount $S_2 - R^*$ to the fundraising campaign, which yields an expected outcome of more than $S_2 - R^*$, so that the numerator in equation (9) is larger than S_2. Second, the marginal reduction of utility due to a decrease in the size of the public good is relatively lower for version two. Consequently, the utility loss because of $G_i(\overline{F}) < G_i(F_i^*)$, $i = 1, 2$ is relatively larger in the first project variant, which makes it relatively less attractive.

Depending on the arrangement of $\overline{F}$, S_2 and β, the fundraising cap results in a breach of inequality (9) and, thus, in a choice of project version two with an observed fundraising share being larger than $\overline{F}$. This scenario is depicted in figure 1. In the most drastic case with a sufficiently high service fee income ($S_2 \geq F_i^* - \overline{F}$), the final fundraising expenditures remain completely unaffected by the regulation and the size of the public good is larger than in the status quo. Solely, the utility of the entrepreneur from providing the public good is significantly lower, which indicates a departure from the social mission.

4 Conclusion

The previous analysis showed how the lead donor's efforts to restrict the entrepreneur's fundraising expenditures can negatively affect the project's mission but, simultaneously, remain ineffective to reduce the level of solicitations itself. This, however, is likely to occur only with

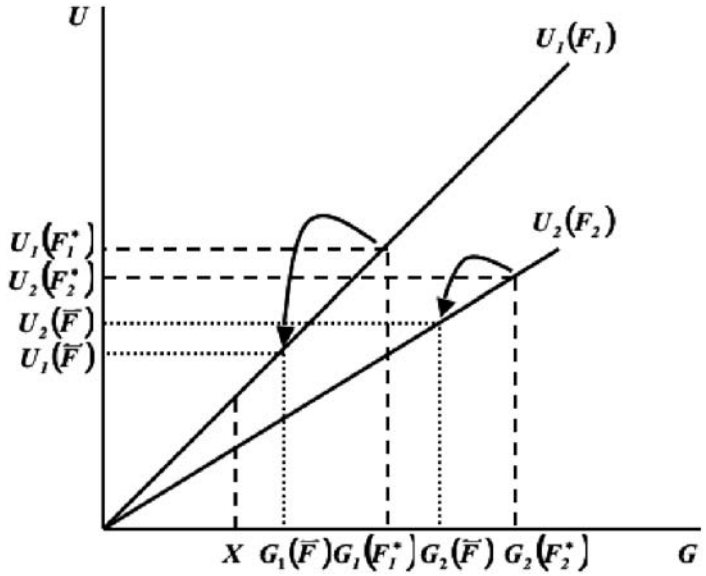


Fig. 1. Effects of a limit on fundraising expenditures

those social projects in which potential service fee revenues are relatively high and the entrepreneurial utility loss from charging the fee is relatively limited. Examples for such cases are not rare, in particular with regard to social entrepreneurship projects in industrialized countries. Indeed, prices of cultural institutions generally vary up to production costs, churches usually claim contributions, and even soup kitchens in European metropolian areas are not for free. Thus, one can observe that recipients are able to afford some expenditure to satisfy their basic needs and the marginal rate of exclusion by charging an additional Euro seems to be sufficiently low. Consequently, potential regulation attempts of the lead donor to restrict the entrepreneur's level of solicitations should largely remain ineffective.

References

1. Andreoni, J. (1998), *Toward a theory of charitable fund-raising*, Journal of Political Economy, 106 (7):28.
2. Andreoni, J. (2006), *Leadership giving in charitable fund-raising*, Journal of Public Economic Theory, 8(1):23.
3. Thornton, J. (2006), *Nonprofit fund-raising in competitive donor markets*, Nonprofit and Voluntary Sector Quarterly, 35: 21.

Part X

Finance, Banking and Insurance

Studying Impact of Decision Making Units Features on Efficiency by Integration of Data Envelopment Analysis and Data Mining Tools

Ali Azadeh and Leili Javanmardi

Department of Industrial Engineering, Faculty of Engineering, University of Tehran, P.O. Box 11365-4563, Iran
aazadeh@ut.ac.ir, javanmardi@ut.ac.ir

Summary. Generally, data mining is the process of analyzing data from different viewpoint and summarizing it into valuable information. This area presents new theories and methods for processing large volumes of data and has obtained noteworthy consideration among researchers. In this paper, a new approach for decision-making process is developed based on the rough set theory of data mining and neural networks combined with data envelopment analysis method. The proposed procedure assesses the effect of personnel attributes on efficiency, utilizing DEA tool in estimating the efficiency of alternative decision making unites. By developing decision system, rough set theory is applied for feature selection (reducts) and all of plausible and meaningful ANN models are constructed for each reduct. Finally DEA method is used for selecting the best reduct and also most important personnel attributes for efficiency analysis. Persian bank branches employed for data generation and the characteristics of its personnel are analyzed on effectiveness of bank branches.

Key words: Data Mining, Data Envelopment Analysis, Rough Set Theory, Artificial Neural Network, Decision Making.

1 Introduction

Too many immeasurable influences and complex relationships among attributes impact efficiency in companies. The rough set, proposed by Z.Pawlak, is one of the techniques for the identification and recognition of common patterns in data [1, 2]. This technique has found too many applications in knowledge discovery. At the present, the study on rough set theory is focusing on feature selection techniques with much success

[3, 4, 5, 6, 7, 8, 9]. However, existing heuristic rough set approaches to feature selection are insufficient at finding optimal reductions. On the other hand, it is not feasible to search for optimal in even average sized datasets. So the combination of this method by other robust data mining tools may help practitioners to go further into feature selection to obtain more accurate results. In this paper, a new approach for feature selection by the combination of rough set theory, neural network and data envelopment is proposed. An effective algorithm is formulated, which will verify the critical attributes influencing efficiency in organizations. Persian bank branches employed for data generation and the characteristics of its personnel are analyzed on effectiveness of bank branches.

2 An Integrated Algorithm for Decision Making Procedure

Efficiency relevant to human attributes is a goal that is rarely questioned in contemporary organizations. As Personnel specifications have greatest impact on efficiency, they can help us designing work environments for maximizing efficiency. The model described in this section proposes an analytic function which predicts these factors exactly. This model is applicable for all problems associated with making decision in companies comprises of decision making units and will be valuable for executive and senior managers. This algorithm has the following general basic steps:

Step 1: Calculate efficiency of each Decision making unit.
Step 2: Determine decision system and collect related information.
Step 3: Data pre-processing.
Step 4: Apply rough set algorithm to determine plausible reducts.
Step 5: Select preferred ANN for each reduct by cross validation.
Step 6: Select best reduct by Data Envelopment Analysis.

3 Case Study

Explanation of implementing every stage of mentioned procedure is described through a case study. The case study focuses on Persian bank branches to analyze the effect of personnel specifications on bank branches efficiency.

3.1 Calculate Efficiency of Each Decision Making Unit

Efficiency scores of bank branches are calculated on major features of production financial firms, containing, number of employees and fixed assets as inputs and loans, deposits and operating income as outputs according to the nature of financial firms in Iran. Related data was obtained for 102 branches of Persian bank for the year of 2006 and Output-oriented BCC Model for efficiency calculation is used to calculate efficiency scores.

3.2 Determine Decision System and Collect Related Information

To construct decision system of this case study, four groups of personnel are identified in each branch. First group are associated as tellers who conduct most of a bank's routine transactions. The second group consists of supervisors who cashing checks and performing controlling task on tellers transactions. Branch mangers and their assistants are in third and fourth groups. Each branch has one branch manager and may have several branch assistants. We have recognized 28 conditional attributes of personnel specifications in this study with the decision attribute of efficiency. Personnel attributes are categorized in four groups of quantity, education, age and work experience as follows:

*Quantity : No. of males,No. of singles,No. of Tellers
*Education :No. of MS, No. of BS,No. of upper Diploma,No. of Diploma, No. of below Diploma
*Age :Average,Minimum,Maximum Age of each personnel group
*Work Experience : Average,Minimum,Maximum Work Experience of each personnel group

3.3 Data Pre-processing

Naturally, we perform the data pre-processing tasks by applying data normalization. Data in decision system are normalized as follows:

*Quantity (divided by) Number of personnel in each branch
*Education (divided by) Number of personnel in each branch
*Work Experience (divided by)Maximum work experience in each branch

3.4 Apply Rough Set Algorithm to Determine Plausible Reducts

In this step we calculate reducts of rough set theory for constructed decision system. The reducts are generated by genetic algorithm [10]. 12 reducts were extracted as follows :

1. Average Work Experience of Supervisor, Average Age of Teller, Number of Male
2. Number of Male, Average Age of Supervisor, Average Age of Teller
3. Number of Male, Number of Teller, Average Work Experience of Assistant
4. Minimum Age of Teller, Maximum Work Experience of Teller, Minimum Work Experience of Assistant
5. Number of Male, Work Experience of Branch manager, Minimum Work Experience of Supervisor
6. Number of Male, Maximum Age of Supervisor, Maximum Work Experience of Supervisor
7. Number of Single, Number of BS, Average Work Experience of Teller
8. Minimum Age of Supervisor, Average Work Experience of Assistant, Maximum Work Experience of Supervisor
9. Number of Single, Maximum Age of Teller, Work Experience of Branch Manager
10. Work Experience of Branch Manager , Minimum Age of Supervisor, Maximum Age of Assistant
11. Average Work Experience of Teller, Number of BS, Number of Single, Number of Male
12. Number of Teller, Number of BS, Number of Single, Number of Male

3.5 Select Preferred ANN for Each Reduct by Cross Validation

Much of the emphasis here is selecting a good subset of conditional features according to different data set of reducts. As all reducts have classification quality of 100% in training data set, we may differentiate their performance by calculating their accuracy in predicting efficiency of unseen objects. For this reason, neural networks are constructed for each reduct. To estimate the quality of constructed neural networks, cross validation test technique is used. As discussed by Cybenko and Patuwo et al., a single hidden layer is sufficient in constructing neural nets [11]. To find the appropriate numbers of hidden nodes in ANN analysis of each reduct, following steps are performed to construct networks with one to q nodes, where q is an optional parameter and will be changed until the desired goal error met by the algorithm.

- Training step, using scaled conjugate gradient training algorithm [12]
- Evaluate the model using the test data and obtaining MAPE error

Where we define,

- Variable MAPEijk is defined as: Mean Absolute Percentage Error of ANN, with regard to kth part of data, used as test data and j node in hidden layer, both related to ith reduct.

- Variable ERRij is defined as: ERRij= Average [MAPEijk: k=1, 2, ..] or average of MAPE in all constructed ANN for ith reduct with regard to j node in hidden layer.

- Variable AERRi is defined as: AERRi =Average (ERRij)

- Variable VarERRi is defined as: VarERRi =Variance (ERRij)

- Variable MaxERRi is defined as: MaxERRi = Max (ERRij)

- Variable MinERRi is defined as: MinERRi = Min (ERRij)

Finally these variables will use to rank the performance of each reduct according to their constructed ANN. In order to calculate error variables,Cross validation test technique is employed with 4 folds. The value of the desired minimum error has been defined between 6 and 8 percent and the value of q has been defined 40. Table 1 shows error variables of MAPEijk calculated for reduct 1 versus number of nodes in hidden layer.

Table 1. Mean Absolute Percentage Error of Reduct 1

Neuron#	40	35	30	25	20	15	10	9	8	7	6	5	4	3	2	1
Run I:	0.440	0.544	0.692	0.119	0.093	0.109	0.160	0.172	0.056	0.070	0.066	0.055	0.060	0.088	0.080	0.028
Run II:	0.777	0.480	0.592	0.388	0.093	0.237	0.225	0.506	0.181	0.157	0.169	0.155	0.153	0.176	0.170	0.195
Run III:	0.111	0.136	0.829	0.416	0.101	0.105	0.103	0.336	0.037	0.053	0.039	0.028	0.083	0.039	0.053	0.047
Run IV:	0.474	0.328	0.675	0.469	0.824	0.288	0.229	0.153	0.358	0.171	0.184	0.177	0.206	0.188	0.190	0.175
ERR1j	0.451	0.372	0.697	0.348	0.568	0.185	0.182	0.267	0.158	0.113	0.115	0.104	0.125	0.123	0.123	0.111

3.6 Select Best Reduct by Data Envelopment Analysis

DEA have been utilized to rank reducts according to error variables.We treated scores of AERRi, VarERRi, MaxERRi and MinERRi for each reduct as inputs of efficiency with the specific output of 1 to calculate efficiency of each reduct. 9th reduct identified as the best one, with gratest full rank efficiency score. By constructing neural network on 9th reduct,we will obtain effective expert systems, which can be utilized by senior manager of the bank for sensitive analysis or efficiency prediction of inefficient or new bank branches according to their personnel specifications values.

4 Conclusion

The methodology proposed in this paper provided a six-stage analysis to help companies formulate an effective decision-making procedure to demonstrate critical factors impacting efficiencies of decision making unites.This approach has been applied to the particular case of Persian bank, evaluating personnel specification impact on bank branches efficiency. The study shows that three futures of 9th reduct from 28 conditional features, have a critical impact on the efficiency of bank branches. This reduction in features number decrease the time of decision making and consequently reduces the cost of efficiency evaluation.

References

1. Z. Pawlak, 1982,Rough sets, International Journal of Computer and Information Sciences, Vol 11, pp. 314-356.
2. Z. Pawlak, 1991, Rough Sets: Theoretical Aspects of Reasoning about Data. Boston, MA: Kluwer.
3. J. Stefanowski, 1997, Rough sets as a tool for studying attribute dependencies in the urinary stones treatment data set, in Rough Sets and Data Mining: Analysis and Imprecise Data, pp. 177-196.
4. Andrew Kusiak, 2000. Autonomous Decision-Making: A Data Mining Approach. 274 IEEE Transactions on Information Technology in biomedicine, Vol 4 Issue 4, pp. 274-284.
5. Weijun Xia, 2007, Supplier selection with multiple criteria in volume discount environments, Omega, Vol 35, Issue 5, October 2007, pp. 494-504.
6. Tzu-Liang, 2007, Rough set-based approach to feature selection in customer relationship management, Omega, Vol 35, Issue 4, pp.365-383.
7. Lian-Yin Zhai,2002, Feature extraction using rough set theory and genetic algorithms-an application for the simplification of product quality evaluation ,Computers & Industrial Engineering, Vol 43, Issue 4, pp.661-676.
8. Xiangyang Wang, 2006, Rough set feature selection and rule induction for prediction of malignancy degree in brain glioma Computer Methods and Programs in Biomedicine, Vol 83, Issue 2, pp.147-156.
9. Andrew Kusiak,2000, Data mining in engineering design: a case study, IEEE International Conference on Robotics and Automation, Proceedings Vol 1,pp.206 - 211.
10. Wróblewski ,1998, Genetic algorithms in decomposition and classification problem, Physica-Verlag, Heidelberg, pp. 471-487.
11. Patuwo E., Hu M.Y., Hung M.S, 1993, Two-group classification using neural networks, Decision Sciences ,Vol 24, Issue 4, pp. 825-845.
12. Moller, M. F., 1993, A scaled conjugate gradient algorithm for fast supervised learning," Neural Networks, Vol. 6, pp. 525-533.

Analysts' Dividend Forecasts, Portfolio Selection, and Market Risk Premia

Wolfgang Breuer[1], Franziska Feilke[2], and Marc Gürtler[2]

[1] Department of Finance, RWTH Aachen University, Germany
wolfgang.breuer@rwth-aachen.de
[2] Department of Finance, Technische Universität Braunschweig, Germany
f.feilke@tu-braunschweig.de, marc.guertler@tu-braunschweig.de

1 Introduction

Today, there is no doubt that modern portfolio selection theory was initiated by the famous contribution of Markowitz [5]. However, even more than fifty years later the practical relevance of Markowitz's work lies far behind the theoretical impact of his ideas. Practical applications of the Markowitz approach are mostly impeded by the necessity of the adequate estimation of expectation values, variances, and covariances of security returns. While historical estimations for variances and covariances work quite satisfactorily, return realizations are only a poor proxy for actual expected future returns. We want to examine the efficiency of portfolio management decisions based on estimated expected returns derived from analysts' dividend forecasts (thereby allowing for non-flat term structures of interest rates and German tax rules) in comparison to nine other portfolio selection strategies. Moreover, we analyze consequences of expectation biases in dividend forecasts and estimate market risk premia.

2 Theoretical Background

We make use of a variant of the multi-period CAPM by Wiese [7]. Consider a firm j with expected dividends $E(\tilde{d}_{j,t})$ according to analysts' forecasts. Let $r_{f,t}$ stand for the riskless interest rate from $t-1$ to t, $\tilde{r}_{j,t}$ for the security returns, and $\tilde{r}_{m,t}$ for market portfolio returns. We introduce $\tau^{(equ)}$ as the tax rate for dividends and capital gains. Moreover, we need $\tau^{(debt)}$ to denote the tax rate for fixed income (debt)

financial instruments. The risk premium $\Phi_{j,t}$ shall be the same for all $t = 1, \ldots, T$. We assume a constant dividend growth rate g_j from $t = 3$ on. Forward rates are available up to $\hat{T} = 15$. We assume $r_{f,t} = r_{f,\hat{T}}$ for $t \geq \hat{T}$. Thus, our adaption of the formula of Wiese [7] is:

$$\begin{aligned} V_{j,0} = & \sum_{t=1}^{3} \frac{E(\tilde{d}_{j,t}) \cdot (1-\tau^{(equ)})^t}{\prod_{\kappa=1}^{t} (1-\tau^{(equ)} + r_{f,\kappa} \cdot (1-\tau^{(debt)}) + \Phi_j)} \\ & + \sum_{t=4}^{15} \frac{E(\tilde{d}_{j,3}) \cdot (1+g_j)^{t-3} \cdot (1-\tau^{(equ)})^t}{\prod_{\kappa=1}^{t} (1-\tau^{(equ)} + r_{f,\kappa} \cdot (1-\tau^{(debt)}) + \Phi_j)} \\ & + \frac{1}{(1-\tau^{(equ)} + r_{f,15} \cdot (1-\tau^{(debt)}) + \Phi_j - (1+g_j) \cdot (1-\tau^{(debt)}))} \\ & \cdot \frac{E(\tilde{d}_{j,3}) \cdot (1+g_j)^{13} \cdot (1-\tau^{(equ)})^{16}}{\prod_{\kappa=1}^{15} (1-\tau^{(equ)} + r_{f,\kappa} \cdot (1-\tau^{(debt)}) + \Phi_j)}. \end{aligned} \tag{1}$$

(1) is to be solved with respect to Φ_j for four different scenarios (see below) in order to derive estimators for expected stock returns.

3 Empirical Setting

We base our empirical examination on monthly data of (all) 16 out of 30 equity shares that belong to the DAX from 01/01/1994 until 07/01/2004. For any point in time t from 12/01/1996 until 06/01/2004 we apply the following portfolio selection strategies (the first four are based on dividend forecasts and the calculation of Φ_j according to formula (1)): (1) flat interest rate structure without German taxation, (2) non-flat term structure without taxes, (3) flat term structure with taxes, (4) non-flat term structure with taxes, (5) market portfolio, (6) equally weighted portfolio, (7) variance minimal portfolio, (8) and (9) portfolios based on historical data (without/with taxes), (10) and (11) three-factor model portfolios according to Fama and French [2] (without/with taxes), (12) and (13) Bayesian portfolios according to Kempf et al. [4] (without/with taxes). We estimate excess return variances and covariances on the basis of historical excess returns with or without taxes. We determine several performance measures: Sharpe ratio, Jensen's alpha, Treynor ratio, and Treynor-Black ratio.[3] For all 13 cases we compute 91 successive optimal portfolios with a time horizon of one

[3] See i.e. [1].

month each subject to short sales constraints. Without loss of generality, we define an arbitrary level of risk of $\sigma_P = 3\%$, and maximize expected excess return. We apply the average of the last 5 years of the annual growth rate of the gross national income just before the point in time when the portfolio selection takes place. In the revolving portfolio optimization process we determine corresponding realized portfolio excess rates of returns after taxes. In order to examine the performance of our strategies for different market settings we compute them separately for two subperiods. While the time period from 12/01/1996 to 08/01/2000 is characterized by rising stock prices, the second period from 09/01/2000 to 06/01/2004 describes a situation with falling ones.

4 Empirical Results

Table 1 presents the Sharpe ratios for the 13 cases. Obviously, portfolio management based on analysts' dividend forecasts performs quite well particularly in comparison to the three benchmark strategies 5, 6, and 7. Moreover, taking non-flat term structures and/or taxes into account actually increases the resulting Sharpe ratio. Furthermore, strategy 4 performs well both in the first period and in the second period. All dividend oriented portfolio selection strategies outperform the market portfolio on a 5 % significance level in the second period, according to the Memmel efficiency test [6]. Strategies 6 and 7 are outperformed by strategies 1 to 4 in the second period at least on a 20 % significance level. No other portfolio selection strategies (from 8 to 13) are able to outperform the three benchmark strategies 5, 6, and 7 on a significance level of 20 % or better.

Our results of Table 1 are verified by the findings for the other performance measures: portfolio strategies based on dividend forecasts are quite advantageous, in particular when based on non-flat term structures and after-tax returns. Once again, dividend based approaches perform particularly well in times of falling stock prices (period 2). Only portfolio strategy 4 is able to reach a positive value of Jensen's alpha on a 10 % significance level in both periods. All other significantly positive values for Jensen's alpha are also only achieved by dividend based portfolio selection strategies. Moreover, portfolio strategy 4 is the only one that – on a 10 % level – implies a significantly higher Treynor ratio than the market portfolio in both subperiods.[4] The same holds

[4] The significance was determined according to the test statistic by Jobson and Korkie [3].

Table 1. Sharpe ratios for thirteen portfolio optimization strategies

		period 1		period 2	
#	strategy	φ_S	rank	φ_S	rank
1	div: flat, without taxes	0.2729	2	0.0650	3
2	div: nonflat, without taxes	0.2426	6	0.0838	2
3	div: flat, with taxes	0.2605	4	0.0492	4
4	div: nonflat, with taxes	0.3923	1	0.0955	1
5	market portfolio	0.2647	3	-0.1652	
6	equally weighted	0.2581	5	-0.0869	
7	variance minimal	0.2107	9	-0.1067	
8	math. hist. without taxes	0.1769	10	0.0182	5
9	math. hist. with taxes	0.1769	11	-0.0253	
10	3-factor-model without taxes	0.1357	12	-0.0296	
11	3-factor-model with taxes	0.1357	12	-0.0365	
12	Bayes without taxes	0.2111	7	-0.0809	
13	Bayes with taxes	0.2111	8	-0.0912	

Table 2. Jensen's alphas and Treynor ratios for thirteen portfolio optimization strategies

	Jensen's alpha				Treynor ratio			
	period 1		period 2		period 1		period 2	
#	φ_J	rank	φ_J	rank	φ_T	rank	φ_T	rank
1	0.00267	3	0.00483	4	0.01703	2	0.00459	3
2	0.00096	5	0.00507	3	0.01587	4	0.00654	2
3	0.00350	2	0.00557	2	0.01635	3	0.00350	4
4	0.00437	1	0.00698	1	0.04322	1	0.00753	1
5	0.00099	4	-0.00248	13	0.01361	6	-0.01038	11
6	0.00094	6	0.00015	9	0.01366	5	-0.00555	9
7	0.00068	7	-0.00129	12	0.01311	7	-0.00832	10
8	-0.00094	12	0.00157	5	0.01046	12	0.00179	5
9	-0.00144	13	0.00090	6	0.01046	13	-0.00255	6
10	-0.00052	10	0.00034	7	0.01070	11	-0.00340	7
11	-0.00079	11	0.00027	8	0.01070	10	-0.00432	8
12	0.00004	9	-0.00065	10	0.01240	8	-0.01271	12
13	0.00006	8	-0.00124	11	0.01240	9	-0.01478	13

true for strategy 4 in comparison to strategy 6 and (on a 20 % significance level) to strategy 7. Once again, all other significantly better performance results in comparison to the market portfolio are related to the also dividend oriented strategies 1 to 3. The dividend strategies also have the best four rankings according to the Treynor-Black ratio.

The good performance of portfolio selection strategies that are based on dividend forecasts may be surprising, as there is an extant literature on the biases in analysts' dividend forecasts. In order to examine the consequences of too optimistic/pessimistic market assessments by analysts, we analyze four additional scenarios with all dividend forecasts adjusted by +10 %, +20 %, –10 %, or –20 % in comparison to the "true" dividend forecast. We find that such a general forecasting bias is not able to considerably alter the resulting ranking of the four dividend oriented strategies in comparison to the other portfolio selection strategies. 16 out of 32 cases have no changes in the ranking, in twelve cases there is a ranking difference of one and in the remaining four cases the ranking difference is greater than one. This may be viewed as an indirect evidence that general expectation biases in analysts' forecasts are only of minor relevance for issues of portfolio management.

Each approach for the estimation of expected security returns may also serve as a means for estimating market risk premia. After estimating individual expected security returns, one simply has to compute the implied expected excess return of the market portfolio for given current security prices. For strategies 1-4 and 8-13 we estimated market risk premia from 12/01/1996 until 07/01/2004. The market risk premia estimators are lowest for strategies 1 and 3 (about 1 to 2 %). Moreover, strategies 1 and 3 are the only ones that do not imply risk premia of or below zero during the period 2 with falling stock market prices. Accordingly, estimates of risk premia vary much more over the two subperiods when looking at the six approaches not based on dividend forecasts which is certainly not very plausible. Nevertheless, strategies 2 and 4 are also based on dividend forecasts, but lead to negative estimators for market risk premia in both subperiods (from -0.003 to -0.016). But in contrast, strategy 4 is the one with the best overall performance according to Tables 1 and 2. Such a constellation verifies the contradiction between applying a method of expectation formation for portfolio optimization on the one side and for assessing market risk premia on the other side. This is not too surprising, because investors are not really acting according to analysts' dividend forecasts, because in such a

situation of homogeneous expectations everyone would hold the market portfolio. This is not the case, because strategy 4 does not result in a reproduction of strategy 5. As a minimum requirement for reasonable estimates of market risk premia, one has to choose a scenario that is consistent with market equilibrium.

5 Conclusion

Even after fifty years of intense research it still remains quite difficult to design "active" portfolio management strategies that are able to beat "passive" approaches like simply holding the market portfolio. The main objective of this paper was to examine how analysts' dividend forecasts might be utilized for portfolio management purposes. In our empirical section we showed the superiority of portfolio selection strategies based on analysts' dividend forecasts over alternative approaches. Moreover, we found out that biased dividend forecasts are of only minor relevance for the efficiency of the dividend strategies. We demonstrated that superior performance results are not in line with the conjecture that analysts' dividend forecasts are helpful in calculating market premia. From all these findings, we conclude that analysts' dividend forecasts are indeed helpful in portfolio optimization, but not for the estimation of market risk premia.

References

1. Breuer W, Gürtler M, Schumacher F (2004) Portfoliomanagement I. Gabler, Wiesbaden
2. Fama EF, French KR (1993) Common risk factors in the returns on stocks and bonds. Journal of Financial Economics 33:3–56
3. Jobson JD, Korkie BM (1981) Performance hypothesis testing with the Sharpe and Treynor measure. The Journal of Finance 36:889–908
4. Kempf A, Kreuzberg K, Memmel C (2002) How to incorporate estimation risk into Markowitz optimization. Operations research proceedings 2001 175–182
5. Markowitz HM (1952) Portfolio Selection. Journal of Finance 7:77–91
6. Memmel C (2003) Performance hypothesis testing with the Sharpe ratio. Finance Letters 1:21–23
7. Wiese J (2007) Das Nachsteuer-CAPM im Mehrperiodenkontext — Replik zu Rapp/Schwetzler. Finanzbetrieb 9:116–120

A Two-Stage Approach for Improving Service Management in Retail Banking

Gül Gökay Emel and Çagatan Taskin

Department of Business Administration, Uludag University, Bursa, Turkey
ggokay@uludag.edu.tr, ctaskin@uludag.edu.tr

Summary. Efficiency analysis is a survival tool for bank managers due to severe competition in retail banking industry. Certainly, it is important to identify the inefficient branches of a bank in building better service management and marketing strategies. The problem here is to eliminate the causes of their inefficiencies in order to improve the inefficient branches. In this study, a two-stage methodology consisting of Data Envelopment Analysis (DEA) and data mining is proposed for solving the problem. Firstly, DEA is used for measuring the relative efficiencies of a bank's branches. Secondly, data mining is used to extract useful information from given data consisting of the characteristics of bank branches, thus allowing bank managers to identify the underlying causes of inefficiencies and helping them to improve their service management. The approach is used in the valuation of the one's branches of the biggest banks in Turkey.

Key words: Data Envelopment Analysis, Data Mining, Finance and Banking

1 Introduction

The service industry has emerged as the fastest growing industry compared to the other economies. Efficiency is a significant issue in the service industry. In current economic systems, efficiency analyses have become an important decision support tool for the banking industry. Bank managers want to identify and eliminate the underlying causes of inefficiencies, thus helping their firms to gain competitive advantages or, at least, meet the challenges from others [11]. There exist a large number of earlier parametric and non-parametric studies on the efficiency of bank branches. DEA is an important method of the efficiency analysis. In spite of this, the results of a DEA do not give information about the differing characteristics of efficient and inefficient branches.

The aim of this paper is to propose a two-stage methodology for identifying and improving relatively inefficient bank branches. The integrated form of DEA and data mining is used here to compensate each other. Two-stage methodology includes the use of data envelopment analysis, CCR (Charnes, Cooper and Rhodes) model, and association rule mining method. Firstly, CCR model is implemented to evaluate the relative operational efficiencies of the bank branches from a production perspective [7]. Then, Apriori association rule mining algorithm is employed to discover useful knowledge from the data which consists of the characteristics of efficient and inefficient bank branches in order to guide the improvement of inefficient branches.

2 Research

2.1 Data Envelopment Analysis and Association Rule Mining

DEA provides a nonparametric methodology for evaluating the efficiency of each comparable decision making units (DMUs) relative to one another. An important feature of DEA is its capability to provide efficiency scores, while taking into account of both multiple inputs and multiple outputs [4],[3],[2],[11]. DEA employs linear programming techniques to establish an input-output based efficiency ratio. It also examines the effects of changes in input and output values. One of the fundamental assumptions of DEA is that of a functional similarity of the DMUs. It is assumed that DMUs in the sample of a DEA model are similar if they utilize the same set of inputs and outputs. But in real world, there is diversity in the set of DMUs and DEA does not take the diversity into consideration [8]. DEA model can be divided into an input-oriented and an output-oriented model, depending on the reason for conducting DEA [9],[12].

Inducing association rules is one of the basic tasks of data mining which can be defined as a process of extracting relationships in huge databases [1]. Unlike traditional rule induction which examines one variable at a time, association rules evaluate a combination of variables simultaneously; therefore represent correlated features better [6]. Thus, it can be used as a method for inquiring into the differences between efficient and inefficient DMUs. Literature about bank branch efficiency mostly includes the use of one method alone and pays little attention to the use of DEA and data mining together. Therefore, the integrated use of these two methods will be a significant contribution to the literature.

2.2 Methodology

The population of the study consists of 48 branches of a big bank in Turkey. In the first stage, an output-oriented model of CCR is implemented to evaluate the relative operational efficiencies of 48 branches from a production perspective. Firstly, input and output variables are selected for DMUs after a literature survey [10],[3],[7],[11],[12],[5]. The operational efficiency assessment considers data from January 2007 to June 2007. The input-output set used in this analysis is shown in Table 1, where (t) denotes time period and Δ denotes change in values between the start and the end of period t.

Table 1. Input-Output set

Input Set	**Output Set**
Number of managerial personnel(t)	Δ in personal accounts(t)
Number of clerical personnel(t)	Δ in commercial accounts(t)
Number of computers(t)	Δ in savings accounts(t)
Working space(t)	Δ in personal credit applications(t)
	Δ in commercial credit applications(t)

In the second stage, Apriori algorithm is employed to discover useful knowledge from the collected data consisting of the characteristics of efficient and inefficient bank branches in order to guide the improvement of inefficient bank branches. Clementine 8.1 data mining software is used for this task.

2.3 Findings

The results of DEA show that only one branch has an efficiency score equal to one and the other branches have been found to be inefficient. When the characteristics of this efficient branch is examined, it is found that the branch has minimum input and maximum output values just because of its location. So, it is defined as an outlier and DEA is implemented for the 47 branches. The efficiency scores of some branches are shown in Fig. 1. As seen, only 9 out of 47 bank branches have efficiency scores equal to one. In addition, efficiency scores of 11 branches which are the most inefficient ones can be seen in Fig. 1. After the identification of inefficient bank branches, the underlying causes of inefficiencies should be first identified, then eliminated.

That's why, a database including the characteristics of both the efficient and inefficient branches are explored by Apriori algorithm in

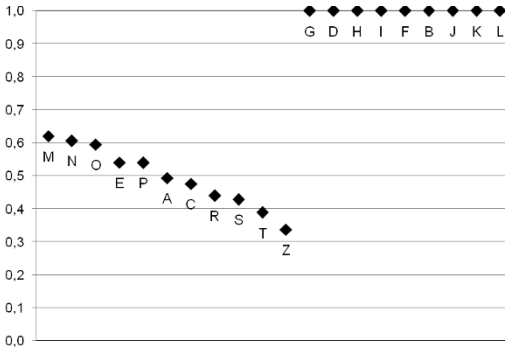


Fig. 1. Efficiency scores of bank branches

Table 2. Sample association rules

Consequent	Antecedents
Branch Name = B (Efficient)	Number of PhD graduates='0' & Open at lunch time ='yes' & Safe-deposit box='yes' & ATM plus='yes'
Branch Name = B (Efficient)	Open at lunch time='yes' & Safe-deposit box='yes' & ATM plus='yes' & Open at weekends='yes'
Branch Name = D (Efficient)	Number of personnel working more than 15 years= '14' & Number of high school graduates='14' & Open at lunch time='yes' & Safe-deposit box='yes'
Branch Name = D (Efficient)	Number of personnel working more than 10 years ='15' & Number of high school graduates='9' & Open at lunch time='yes' & Number of MSc graduates='0'
Branch Name = F (Efficient)	Number of personnel working more than 10 years='11' & Number of high school graduates='8' & Safe-deposit box='yes' & ATM plus='yes'
Branch Name = A (Inefficient)	Number of personnel working more than 15 years='6' & Number of BSc graduates = '10' & Open at lunch time='no' & Safe-deposit box='no'
Branch Name = C (Inefficient)	Open at lunch time='no' & Safe-deposit box='no' & ATM plus='yes' & Number of PhD graduates='2'
Branch Name = E (Inefficient)	Number of personnel working less than 5 years='8' & Number of BSc graduates='12' & Number of high school graduates='3' & Open at weekends='no'
Branch Name = S (Inefficient)	Open at lunch time='no' & Safe-deposit box='no' & ATM plus='yes' & Open at weekends='no'
Branch Name = Z (Inefficient)	Number of personnel working more than 15 years='3' & Number of MSc graduates='5' & Open at lunch time='no' & Safe-deposit box='no'

order to obtain useful knowledge. Some of the extracted association rules are shown in Table 2. As illustrated in Table 2, association rules show the differences between efficient and inefficient branches' characteristics. "Working experience", "open at lunch time", "open at weekends" and "safe-deposit box" variables are found to be important for the efficiency scores of the branches.As seen, working experience of the staff of efficient bank branches is higher than inefficient ones'. Besides, efficient bank branches employ high school graduates more than inefficient branches. It is important for the branch to have a safe-deposit box and working hours of the personnel seem to have a serious effect on the efficiency scores. An association rule has two important measures, support and confidence, which give information about the accuracy of the rule. The results of the rule accuracy analysis are given in Table 3.

Table 3. Rule accuracy analysis

Rules (Efficient Branches)	Support: 11.1%
	Confidence: 100.0%
Rules (Inefficient Branches)	Support: 9.1%
	Confidence: 100.0%

3 Conclusion

Severe competition in banking industry urges banks to be more efficient. So, bank managers should not only identify the inefficient bank branches, but also understand the causes of the inefficiencies. DEA is a powerful method of efficiency analysis, but it only evaluates the efficiency of comparable DMUs relative to one another. Therefore, an integrated form of DEA and data mining is proposed here. In the first stage, DEA is implemented to evaluate the efficiencies of the branches. Then, Apriori association rule mining algorithm is employed to discover the factors that lead to diversity in the DMUs. The findings reflect that there are clear differences between efficient and inefficient branches' characteristics. These differences may help bank managers to guide the improvement of inefficient branches and lead to the formulation of service management strategies. However, this research work has limitations like many others. In the study, the relative operational efficiencies of the bank branches are evaluated only. Future studies should also include the measurement of quality and profitability efficiencies with the help of the proposed two-stage approach. The results of these

efficiency analyses will support bank managers in their service management and marketing decisions.

References

1. Angiulli F., Ianni G., Palopoli L. (2004) On the Complexity of Inducing Categorical and Quantitative Association Rules. Theoretical Computer Science 314: 217-249.
2. Asmild M, Paradi JC, Reese DN, Tam F (2007) Measuring Overall Efficiency and Effectiveness Using DEA. European Journal of Operational Research 178: 305-321.
3. Cook WD, Zhu J (2007) Classifying Inputs and Outputs in Data Envelopment Analysis. European Journal of Operational Research 180: 692-699.
4. Cooper WW, Seiford LM, Tone K (2006) Introduction to Data Envelopment Analysis and Its Uses. Springer-Verlag, USA.
5. Das A, Ray SC, Nag A (2007) Laboruse Efficiency in Indian Banking: A Branch Level Analysis. Omega, Article in Press.
6. Iglesia B. De La, Richards G., Philpott M.S., Rayward-Smith V.J. (2006) The Application and Effectiveness of a Multi-objective Metaheuristic Algorithm for Partial Classification. European Journal of Operational Research 169: 898-917.
7. Portela MCAS, Thanassoulis E (2007) Comparative Efficiency Analysis of Portugese Bank Branches. European Journal of Operational Research 177: 1275-1288.
8. Samoilenko S, Osei-Bryson K-M (2007) Increasing the Discriminatory Power of DEA in the Presence of the Sample Heterogeneity with Cluster Analysis and Decision Trees. Expert Systems with Applications, doi:10.1016j.eswa.2007.01.039.
9. Seol H, Choi J, Park G, Park Y (2006) A Framework for Benchmarking Service Process Using Data Envelopment Analysis and Decision Tree. Expert Systems with Applications, Article in Press.
10. Valverde SC, Humphrey DB, Paso RLd (2007) Do Cross-Country Differences in Bank Efficiency Support a Policy of "National Champions"?. Journal of Banking & Finance 31: 2173-2188.
11. Wu DD, Yang Z, Liang L (2006a) Efficiency Analysis of Cross-Region Bank Branches Using Fuzzy Data Envelopment Analysis. Applied Mathematics and Computation 181: 271-281.
12. Wu DD, Yang Z, Liang L (2006b) Using DEA-neural Network Approach to Evaluate Branch Efficiency of a Large Canadian Bank. Expert Systems with Applications 31: 108-115.

Non-maturing Deposits, Convexity and Timing Adjustments

Oliver Entrop and Marco Wilkens

Catholic University of Eichstätt-Ingolstadt
oliver.entrop@ku-eichstaett.de, marco.wilkens@ku-eichstaett.de

1 Introduction

One key driver of a bank's total interest rate risk is the position of non-maturing deposits. Several papers such as [6], [8], and [7] value non-maturing deposits in an arbitrage-free framework and analyze their risk profile. All these models consist of three major components: first, the short rate process, i.e. the dynamics of the default-free interest rate term structure; second, the interest rate pass-through, i.e. the link between the development of the deposit rates and the development of default-free interest rates, in general the short rate; third, the development of the deposit volume over time. In this paper, we concentrate on the interest rate pass-through. We provide some term structure model-free results on the valuation of deposits, when the deposit rates are linearly linked to some long-term swap rate (rather than a short-term interest rate) as the reference rate with an unnatural time lag.

2 Deposits

2.1 Preliminaries

Let $(\Omega, (F_t)_{t\in\{0,T'\}}, P)$ be a filtered probability space that fulfills the usual conditions. Like the aforementioned articles, we assume markets to be arbitrage-free, frictionless and complete. Let $P(t,T), 0 \leq t \leq T \leq T'$, denote the value in t of a default-free zero bond with face value 1 maturing in T. The filtration is assumed to be generated by these zero bonds. The M-year swap rate $SR(t,M)$, $M \in \mathbb{N}$, in t and the corresponding today's M-year forward swap rate $FSR(t,M)$ are given by

$$SR(t,M) = \frac{1 - P(t,t+M)}{\sum_{j=1}^{M} P(t,t+j)}, \tag{1}$$

$$FSR(t,M) = \frac{P(0,t) - P(0,t+M)}{\sum_{j=1}^{M} P(0,t+j)}. \tag{2}$$

Q^t denotes the unique equivalent t-forward martingale measure (see [3]) and $E_0^t()$ the respective expectation operator conditional on today. Under Q^T, the value of a non-dividend paying security discounted by $P(t,T)$ is a martingale. As a special case, the expected value for the point in time t of a zero bond maturing in T under the t-forward measure equals its forward price:

$$E_0^t(P(t,T)) = \frac{P(0,T)}{P(0,t)}. \tag{3}$$

2.2 Valuation

Define $0 = t_0, t_1, t_2, ..., t_N = T$ with $t_i - t_{i-1} = \Delta t = 1$.[1] For simplicity, we consider a deposit with a constant face value 1 and maturity date T. The deposit rate for the period $]t_{i-1}, t_i]$ paid in t_i is given by $DR(t_{i-1})$. We assume that $DR(t_{i-1})$ is fixed at $t_{i-1}^k := t_{i-1} + k$ with $-1 \leq k \leq 1$. The 'shift' k has a straightforward interpretation: if $k = 0$ the deposit rate is fixed at the beginning of the period $]t_{i-1}, t_i]$. In this case we have a 'natural' time lag between the fixing date and the date at which the deposit rate is paid. If $k < 0$ the deposit rate is fixed before the beginning of the respective period. If $k > 0$ it is fixed within the period. In both cases there is an 'unnatural' time lag. We assume that the deposit rate for the period $]t_{i-1}, t_i]$ is linearly linked to the M-year swap rate $SR(t_{i-1}^k, M)$ observed in t_{i-1}^k as the reference rate:[2]

$$DR(t_{i-1}) = b_1 + b_2 \; SR(t_{i-1}^k, M). \tag{4}$$

By construction, $DR(t_{i-1})$ is measurable with respect to F_{t_i}. It can be interpreted as a European derivative on the term structure that is due in t_i. Therefore, we obtain the following representation of the present value PV of the deposit:

$$PV = \; b_1 \sum_{i=1}^{N} P(0,t_i) + P(0,t_N) + b_2 \sum_{i=1}^{N} P(0,t_i) \; E_0^{t_i}(SR(t_{i-1}^k, M)). \tag{5}$$

[1] The assumption $\Delta t = 1$ can easily be relaxed.

[2] Obviously, the deposit is close to a portfolio consisting of a money market floater and a constant maturity swap. See, e.g., [1] and [4] for the valuation of constant maturity swaps.

Clearly, the key to the calculation of PV is the determination of the present value of $SR(t_{i-1}^k, M)$ paid in t_i as the other components of (5) can be calculated easily. Define

$$PV(SR(t_{i-1}^k, M), t_i) = P(0, t_i)\ E_0^{t_i}(SR(t_{i-1}^k, M)). \tag{6}$$

In the following, we aim to calculate an adjustment $AD(t_i)$ on the respective forward swap rate, so that

$$PV(SR(t_{i-1}^k, M), t_i) = P(0, t_i)\ (FSR(t_{i-1}^k, M) + AD(t_i)) \tag{7}$$

is fulfilled. As the present value of $SR(t_{i-1}^k, M)$ paid in t_i equals the present value of $P(t_{i-1}^k, t_i)\ SR(t_{i-1}^k, M)$ paid in t_{i-1}^k

$$PV(SR(t_{i-1}^k, M), t_i) = P(0, t_{i-1}^k)\ E_0^{t_{i-1}^k}(P(t_{i-1}^k, t_i)\ SR(t_{i-1}^k, M)) \tag{8}$$

must hold. By equating (6) and (8) and using the definition of the covariance we obtain[3]

$$\begin{aligned} E_0^{t_i}(SR(t_{i-1}^k, M)) &= \frac{P(0, t_{i-1}^k)}{P(0, t_i)}\ E_0^{t_{i-1}^k}(P(t_{i-1}^k, t_i)\ SR(t_{i-1}^k, M)) \\ &= \frac{P(0, t_{i-1}^k)}{P(0, t_i)}\ E_0^{t_{i-1}^k}(P(t_{i-1}^k, t_i))\ E_0^{t_{i-1}^k}(SR(t_{i-1}^k, M)) \\ &\quad + \frac{P(0, t_{i-1}^k)}{P(0, t_i)}\ CoVar_0^{t_{i-1}^k}(P(t_{i-1}^k, t_i), SR(t_{i-1}^k, M)). \end{aligned} \tag{9}$$

Substituting (3) into (9) and rearranging terms leads to

$$\begin{aligned} &E_0^{t_i}(SR(t_{i-1}^k, M)) \\ &= E_0^{t_{i-1}^k}(SR(t_{i-1}^k, M)) \\ &\quad + \frac{P(0, t_{i-1}^k)}{P(0, t_i)}\ CoVar_0^{t_{i-1}^k}(P(t_{i-1}^k, t_i), SR(t_{i-1}^k, M)) \\ &= FSR(t_{i-1}^k, M) + CA(t_i) + TA(t_i), \end{aligned} \tag{10}$$

where

$$CA(t_i) = E_0^{t_{i-1}^k}(SR(t_{i-1}^k, M)) - FSR(t_{i-1}^k, M), \tag{11}$$

$$TA(t_i) = \frac{P(0, t_{i-1}^k)}{P(0, t_i)}\ CoVar_0^{t_{i-1}^k}(P(t_{i-1}^k, t_i), SR(t_{i-1}^k, M)). \tag{12}$$

[3] The first line of (9) is a special case of the change of numéraire theorem, see [3].

Equations (6), (10), (11), and (12) clarify that the forward swap rate has to be adjusted by two terms: first, by the difference $CA(t_i)$ between the t_{i-1}^k-forward measure expectation of the swap rate and the corresponding forward swap rate. Second, by the scaled covariance $TA(t_i)$ between the discount factor from t_i to t_{i-1}^k and the swap rate under the t_{i-1}^k-forward measure. The two adjustments sum up to the total adjustment $AD(t_i) = CA(t_i) + TA(t_i)$.

2.3 Convexity and Timing Adjustment

We first analyze the so called 'convexity adjustment' $CA(t_i)$. Based on the definition of the covariance and of the swap rate (1) we obtain

$$\begin{aligned} &E_0^{t_{i-1}^k}(SR(t_{i-1}^k, M)) \\ &= E_0^{t_{i-1}^k}\left(\frac{1}{\sum_{j=1}^{M} P(t_{i-1}^k, t_{i-1}^k + j)}\right) E_0^{t_{i-1}^k}(1 - P(t_{i-1}^k, t_{i-1}^k + M)) \\ &\quad + CoVar_0^{t_{i-1}^k}\left(\frac{1}{\sum_{j=1}^{M} P(t_{i-1}^k, t_{i-1}^k + j)}, 1 - P(t_{i-1}^k, t_{i-1}^k + M)\right). \end{aligned} \tag{13}$$

As the function $x \to 1/x$ is convex Jensen's inequality, (3) and (2) imply

$$\begin{aligned} &E_0^{t_{i-1}^k}\left(\frac{1}{\sum_{j=1}^{M} P(t_{i-1}^k, t_{i-1}^k + j)}\right) E_0^{t_{i-1}^k}(1 - P(t_{i-1}^k, t_{i-1}^k + M)) \\ &\geq \frac{1}{\sum_{j=1}^{M} E_0^{t_{i-1}^k}(P(t_{i-1}^k, t_{i-1}^k + j))} E_0^{t_{i-1}^k}(1 - P(t_{i-1}^k, t_{i-1}^k + M)) \\ &= \frac{1}{\sum_{j=1}^{M} \frac{P(0, t_{i-1}^k + j)}{P(0, t_{i-1}^k)}} \frac{P(0, t_{i-1}^k) - P(0, t_{i-1}^k + M)}{P(0, t_{i-1}^k)} \\ &= FSR(t_{i-1}^k, M). \end{aligned} \tag{14}$$

As the covariance term in (13) is positive in general, we obtain the following inequality for the expected swap rate:

$$E_0^{t_{i-1}^k}(SR(t_{i-1}^k, M)) \geq FSR(t_{i-1}^k, M). \tag{15}$$

This implies that the convexity adjustment $CA(t_i)$ is positive in general. Note that it only depends on the fixing date t_{i-1}^k and not on the

payment date t_i of the deposit rate. Therefore, it is also independent of the difference between these two dates, i.e. the time lag. The convexity adjustment would also be necessary if the deposit rate fixed in t_{i-1}^k were paid in t_{i-1}^k rather than in t_i.

In contrast, the second adjustment $TA(t_i)$ depends only on the difference between the fixing date and the payment date, i.e. the time lag. Therefore, it is called 'timing adjustment'.[4] If the time lag is zero, i.e. if $k = 1$, we have $P(t_{t-1}^k, t_i) = 1$ in (12). Therefore, the covariance term equals zero so that the timing adjustment vanishes, i.e. $TA(t_i) = 0$. Since generally all interest rates are positively correlated the covariance in (12) and, hence, the timing adjustment is negative for $k < 1$. Note that the timing adjustment is even necessary if we have a natural time lag, i.e. if $k = 0$.

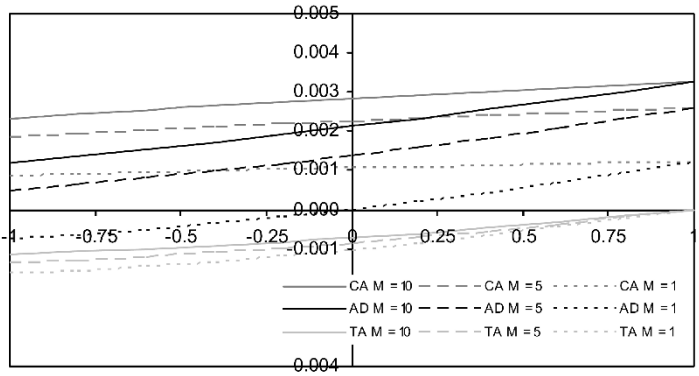


Fig. 1. Convexity and timing adjustments

This figure shows numerical results for the convexity adjustment CA, the timing adjustment TA and the total adjustment AD for different maturities M of the swap rate in dependence of the time lag k. The adjustments are calculated for $t_i = 5$. Calculations are based on the short rate model of Hull and White (see [5]) with the following input parameters: today's spot rate structure = flat at 5%; mean reversion speed = 0.1; short rate volatility = 0.02

Of course, the concrete size of the convexity adjustment, the timing adjustment, and the total adjustment depends on the term structure model. Figure 1 shows some numerical results for the model by Hull and White (see [5]). The timing adjustment equals zero for $k = 1$ and

[4] See also [4].

is becoming negative and smaller for smaller k. The convexity adjustment is always positive and is becoming larger for larger k. The total adjustment is increasing in k and is positive in general. For $k < 0$ and small maturities M of the reference swap rate it can become negative. Obviously, most effects are more pronounced for longer maturities of the swap rate.

3 Concluding Remarks

In this paper, we analyzed the valuation of deposits when the deposit rates are linearly linked to long-term swap rates. We allowed for natural and unnatural time lags and provided term structure model-free results on the valuation of these deposits. We especially focused on the structure of necessary adjustments on the forward swap rates: the convexity and the timing adjustment. Our analysis can easily be transferred to the case of other capital market yields such as spot rates or yields of fixed-coupon bonds (see [2]).

References

1. Brigo D, Mercurio F (2001) Interest Rate Models: Theory and Practice. Springer, Berlin Heidelberg New York
2. Entrop, O (2007) Einlagenbewertung und Einlagensicherung in Banken: Ein Beitrag zum Kapitalmarktorientierten Bankmanagement im strukturmodelltheoretischen Kontext. Berliner Wissenschafts-Verlag, Berlin, forthcoming
3. Geman H, El Karoui N, Rochet JC (1995) Changes of Numéraire, Changes of Probability Measure and Option Pricing. Journal of Applied Probability 32:443–458
4. Hull JC (2003) Options, Futures and Other Derivatives. Prentice Hall, Upper Saddle River
5. Hull JC, White A (1990) Pricing Interest-Rate-Derivative Securities. Review of Financial Studies 3:573–592
6. Jarrow RA, van Deventer DR (1998) The arbitrage-free valuation and hedging of demand deposits and credit card loans. Journal of Banking and Finance 22:259–272
7. Kalkbrener M, Willing J (2004) Risk management of non-maturing liabilities. Journal of Banking and Finance 28:1547–1568
8. O'Brien JM (2000) Estimating the Value and Interest Rate Risk of Interest-Bearing Transactions Deposits. Working Paper, Board of Governors of the Federal Reserve System, November 2000

Nichtparametrische Prädiktorselektion im Asset Management

Johannes Hildebrandt and Thorsten Poddig

Universität Bremen, Wirtschaftswissenschaft, Lehrstuhl für ABWL, insbesondere Finanzwirtschaft
jhi@uni-bremen.de, poddig@uni-bremen.de

1 Einführung

Im Asset Management stellt sich dem Investor die zentrale Aufgabe der systematischen Verteilung des Anlagekapitals auf die verschiedenen Investitionsalternativen. Das moderne Asset Management basiert auf der Portfolio Selection nach Markowitz und damit auf der Annahme gegebener zukünftiger Rendite- und Risikoerwartungen bzgl. der Anlagealternativen. Diese Annahme impliziert ein theoretisch wie praktisch bedeutsames Prognose- bzw. Schätzproblem.

Einige empirische Befunde legen nahe, dass für den Zweck des Asset Managements Renditeschätzungen bedeutsamer als Risikoschätzungen sind (vgl. z.B. [2]). Während für Renditeprognosen mehr oder weniger elaborierte Modelle zum Einsatz kommen, werden als Risikoschätzer häufig noch (gewichtete) empirische Varianz-Kovarianz-Matrizen verwendet (vgl. [1]). Der Zielkonflikt zwischen akzeptiertem Risiko und erwarteter Rendite ist jedoch eines der grundlegensten Probleme in der Finanzwirtschaft, so dass die durch (Ko-)Varianzen gemessene Unsicherheit für die finanzwirtschaftliche Theorie und Praxis eine große Rolle spielt.

Der bedingten Heteroskedastizität von Kapitalmarktrenditen tragen das ARCH-Modell und seine vielfältigen Erweiterungen Rechnung. Nach einer Idee von Petersmeier (vgl. [6]) lassen sich alternativ mit nichtparametrischen Kernregressionsschätzern Renditen und (Ko-)Varianzen in einem integrierten Modell prognostizieren, ohne sich an eine Parametrisierung oder ausschliesslich autoregressive Prädiktoren zu binden. Die inkonsistente Zusammenführung von Prognosen unterschiedlicher Rendite- und Risikomodelle entfällt.

Gegenstand dieses Beitrags ist die Einordnung dieser Idee in den Rahmen der ARCH-Modelle sowie eine adäquate Formulierung der Idee in einem integrierten ökonometrischen Modell. Dies umfasst die Untersuchung der theoretischen Eignung, praktischen Umsetzbarkeit und empirischen Leistungsfähigkeit insbesondere der nichtparametrischen Modellierung der bedingten Heteroskedastizität. Ein solches integriertes nichtparametrisches Rendite- und Risikoprognosemodell besitzt aufgrund seiner Flexibilität und Automatisierbarkeit auch praktische Relevanz für das quantitative Asset Management.

Nach einer Darstellung der Methodik im zweiten Kapitel beschreibt das dritte Kapitel eine empirische Untersuchung. Eine Zusammenfassung schließt den Beitrag ab.

2 Nichtparametrische Prädiktorselektion und Kernregressionsschätzer

Ein ökonometrisches Modell formuliert einen Zusammenhang zwischen ökonomischen Variablen, indem eine erklärte (endogene) Variable in einen funktionalen Zusammenhang zu einer oder mehreren erklärenden (exogenen) Variablen gesetzt wird. Die Auswahl von erklärenden Variablen erfolgt i.d.R. abhängig von der übrigen Modellspezifikation, man entscheidet sich daher zuerst für eine Modellgleichung und selektiert dann die bei Verwendung dieser Modellgleichung relevanten Einflussgrößen.

Der funktionale Zusammenhang ist selten a priori bekannt. Es muss daher entweder eine Funktion angenommen oder der Zusammenhang ohne Parametrisierung rein aufgrund von beobachteten Daten der relevanten Variablen geschätzt werden. Der wesentliche Vorteil nichtparametrischer Modelle besteht in der Vermeidung restriktiver Annahmen, z.B. der Annahme linearer Zusammenhänge oder spezieller Verteilungen.

Ein geeignetes nichtparametrisches Modell ist der sog. Kernregressionsschätzer. Man verwendet lokale gewichtete Mittelwerte als Schätzfunktion (vgl. [3] S. 14f). Durch die entsprechende ökonometrische Modellierung in (1) erhält man zu gegebenen Werten der Einflussgrößen eine Schätzung der Zielgröße durch einen anhand der Distanz der Einflussgrößen gewichteten Durchschnitt früherer Beobachtungen. Der sog. Nadaraya-Watson-Schätzer gibt eine erwartungstreue Schätzung für den bedingten Erwartungswert der abhängigen Variablen über gewichtete Durchschnitte bisheriger Beobachtungen an:

$$\hat{E}(y|x) = \sum_{i=1}^{N} \omega_i(x) \cdot y_i \tag{1}$$

wobei

$$\omega_i(x) = \frac{K\left(\frac{\|x-x_i\|}{h}\right)}{\sum_{j=1}^{N} K\left(\frac{\|x-x_j\|}{h}\right)} \tag{2}$$

Hier bezeichnet x den Vektor der jeweils aktuell betrachteten Einflussgrößen, x_i $(i = 1, ..., N)$ enthält die $i$te Beobachtung der Einflussgrößen, K steht für die geeignet zu wählende sog. Kernfunktion und h ist eine an die beobachteten Daten geeignet anzupassende Bandbreite. Zulässige Kernfunktionen sind nicht negativ und ordnen den kleinsten Distanzen die höchsten Gewichte zu. Gängig ist z.B. eine exponentielle Kernfunktion $K(z) = \exp(-z^2/2)$. Auf Basis der beobachteten Schätzdaten und der herangezogenen Kernfunktion wird die Bandbreite durch ein Optimierungskalkül so gewählt, dass der Schätzfehler minimal ist (vgl. [3]).

Die Gewichte ω_i ergeben sich in Abhängigkeit von Kernfunktion und Bandbreite aus der Distanz des aktuellen x von den beobachteten Werten x_i. Je kleiner die Distanz, desto größer wird das Gewicht der entsprechenden Beobachtung y_i (vgl. (1)). Es werden im Allgemeinen nur die nächsten Nachbarn berücksichtigt, da die Gewichte weiter entfernter Beobachtungen sehr klein werden. Die Berechnung einer Schätzung lässt sich daher anhand der Gewichtung ω_i nachvollziehen.

Den Nadaraya-Watson-Schätzer haben u.a. Pagan und Schwert zur Formulierung eines nichtparametrischen ARCH-Modells verwendet (vgl. [5]). Da die Asset Allokation auf Renditen und (Ko-)Varianzen beruht, liegt die Idee nahe, diese beiden Größen integriert in einem Prognosemodell zu schätzen (vgl. [6] S. 120-123). Dies führt zu einer nichtparametrischen Risikomodellierung unter Berücksichtigung exogener Variablen. Die bedingte Varianz einer betrachteten Größe berechnet sich als Erwartungswert der quadrierten Abweichungen vom bedingten Erwartungswert. Dessen Schätzung erfolgt mittels des Nadaraya-Watson-Schätzers analog zu (1):

$$\widehat{\mathrm{Var}}(y|x) = \hat{E}\left((y - \hat{E}(y|x))^2\right) = \hat{E}(r^2|x) = \sum_{i=1}^{N} \omega_i(x) \cdot r_i^2, \tag{3}$$

wobei $r = y - \hat{E}(y|x)$ die Residuen des Renditeprognosemodells bezeichnet. Für die bedingte Kovarianz zweier Zielgrößen y_1 bzw. y_2 und deren Residuen r_1 bzw. r_2 gilt $\widehat{\mathrm{Cov}}(y_1, y_2|x) = \hat{E}(r_1 \cdot r_2|x)$.

Für dieses Risikomodell benötigt man die Residuen des Renditemodells. Insbesondere modelliert man wie bei ARCH-Modellen als Risikogröße die als Residuen bezeichneten Abweichungen der wahren Werte von den jeweiligen Renditeerwartungswertschätzungen anstatt vom empirischen Mittelwert, der bei der Berechnung der empirischen (Ko-)Varianz heranzuziehen wäre. Die Risikomodellierung und damit die resultierenden Prognosen sind daher im Rahmen des Asset Managements stets im Verbund mit den berücksichtigten Renditeprognosen zu verwenden.

Die Aufgabe der Auswahl geeigneter Prädiktoren aus einer Menge exogener Variablen ist ein zentrales Forschungsfeld nichtparametrischer Prognosemodelle. Lavergne und Vuong (vgl. [4]) haben einen Test vorgeschlagen, dessen Teststatistik es erlaubt, die Signifikanz des Einflusses einer zusätzlichen Variable $x^{(p+1)}$ auf ein vorhandenes Kernregressionsmodell $E_{\{x^{(1)},\ldots,x^{(p)}\}}(y|x)$ mit p Prädiktoren zu berechnen. Die Nullhypothese

$$H_0 : \; E_{\{x^{(1)},\ldots,x^{(p)}\}}(y|x) = E_{\{x^{(1)},\ldots,x^{(p)},x^{(p+1)}\}}(y|x) \tag{4}$$

des statistischen Tests postuliert daher die Gleichheit der beiden Modelle, eine Ablehnung der Nullhypothese impliziert mit hoher Wahrscheinlichkeit einen tatsächlich vorhandenen Einfluss des zusätzlichen Prädiktors $x^{(p+1)}$. Dieser Signifikanztest lässt sich sowohl auf das Renditemodell (1) als auch auf das Risikomodell (3) anwenden, um eine Prädiktorselektion durchzuführen.

3 Empirische Untersuchungen

Zur Untersuchung der Leistungsfähigkeit der Methodik soll aufgrund historischer Daten eine Fallstudie zum Asset Management vorgestellt werden. Die historischen Daten der endogenen Variablen bestehen aus Aktien- (Welt, Deutschland sowie Emerging Markets) und Bondindizes (Europa und USA), als exogene Variablen stehen neben diesen und weiteren Aktien- und Bondindizes auch Zins- und Wechselkursänderungen, Konjunkturdaten sowie Rohstoffdaten zur Verfügung. Alle Daten liegen als Monatsschlusskurse des Zeitraums 12/1998 – 01/2007 vor. Die stetigen Renditen des Zeitraums 01/1999 – 01/2005 bilden die Schätzmenge, aufgrund der pseudo-ex-ante Prognosen für den Zeitraum 02/2005 – 01/2007 wird die Prognoseleistung validiert. Die Schätzungen wurden dynamisch mit rollierenden Zeitfenstern durchgeführt. Die exogenen Variablen wurden um ein bis einschließlich drei Monate zeitverzögert berücksichtigt.

Zu Vergleichszwecken wurden als Renditeprognosemodelle eine multivariate lineare Regression sowie der rollierende empirische Mittelwert und eine naive Prognose als einfache Benchmarkmodelle berücksichtigt. Zum Vergleich der Risikoschätzung wurde ein GARCH-Modell herangezogen. Die signifikanztestbasierte Kernregressionsschätzung liefert die beste Prognosegüte in Bezug auf den über alle Märkte gemittelten Anteil des mitgenommenen Renditepotenzials (Wegstrecke). Bei den Risikoprognosen zeigt sie den kleinsten Prognosefehler (RMSE) und das höchste Bestimmtheitsmaß R^2 (vgl. Tab. 1).

Table 1. Mittlere Prognosegüte 02/2005–01/2007

Renditemodell	RMSE	Trefferquote	Wegstrecke
Kernregr.	3,45 %	0,60	0,32
Lin. Reg.	4,58 %	0,55	0,12
naiv	4,40 %	0,53	0,11
hist. MW	3,21 %	0,60	0,22
Risikomodell	**RMSE**	**R^2**	
Kernregr.	0,28 %	0,44	
GARCH	0,34 %	0,14	
hist. Var.	0,39 %	0,03	

Der kombinierte Effekt der Berücksichtigung der Rendite-und Risikoprognosen wird mittels der Renditedifferenz des jeweils aufgrund der Standard-Markowitz-Optimierung mit Risikoaversionsparameter $\lambda = 3$ gebildeten Portfolios gegenüber dem gleichgewichtetem Benchmarkportfolio untersucht. Die Kernregression zeigt hier die höchste Überrendite und die beste Sharpe-Ratio. Insbesondere durch die Verwendung der nichtparametrischen Risikoprognosen anstatt der empirischen (Ko-)Varianzmatrix der Residuen in Verbindung mit der Kernregressionsschätzung der Renditen ergab sich eine höhere Rendite bei niedrigerer Volatilität. Die Verwendung des GARCH-Modells erzielte hingegen keine Verbesserung der Performance (vgl. Tab. 2).

4 Zusammenfassung

Die Methode der nichtparametrischen Kernregressionsschätzung in Verbindung mit dem Signifikanztest nach Lavergne und Vuong ist für die

Table 2. Annualisierte Performance 02/2005–01/2007

Renditeprognose	Risikoprognose	Überrendite	Volatilität	Sharpe-Ratio
Kernregr.	Kernregr.	6,45 %	9,18 %	0,70
Kernregr.	hist. (Ko-)Var.	4,71 %	10,87 %	0,43
Lin. Reg.	GARCH	-10,95 %	10,09 %	-1,08
Lin. Reg.	hist. (Ko-)Var.	-11,44 %	9,17 %	-1,25
naiv	hist. (Ko-)Var.	0,06 %	18,84 %	0,00
hist. MW	hist. (Ko-)Var.	-6,47 %	7,91 %	-0,82

Verwendung zur integrierten Schätzung von Rendite- und Risikoparametern im Rahmen des quantitativen Asset Managements theoretisch geeignet und mit gängigen statistischen Softwarepaketen relativ einfach zu implementieren.

Das GARCH-Modell bzw. die Prädiktorselektion der nichtparametrischen Kernregression zeigten für die herangezogenen Monatsrenditen Autokorrelationen in den Residuen bzw. signifikante Zusammenhänge mit den berücksichtigten exogenen Variablen. Die Ergebnisse der Untersuchung zeigen, dass mit dem vorgestellten Modellansatz eine Performancesteigerung erzielt werden konnte.

Detailliertere theoretische Betrachtungen und weitere empirische Untersuchungen der nichtparametrischen Modellierung bedingter Heteroskedastizität müssen zeigen, woraus diese bessere Prognoseleistung resultiert und ob eine solche Verbesserung auch in der Deutlichkeit durchgängig zu erwarten ist.

References

1. Bollerslev T et al. (2006) Volatility and Correlation Forecasting. In: Elliot G, Granger CWJ, Timmermann A (eds) Handbook of Economic Forecasting, 1. North Holland, Amsterdam
2. Chopra VK, Ziemba WT (1993) The Effect of Errors in Means, Variances, and Covariances on Optimal Portfolio Choice. Journal of Portfolio Management 19:6–11
3. Härdle W (1999) Applied nonparametric regression. Cambridge Univ. Press, Cambridge
4. Lavergne P, Vuong Q (2000) Nonparametric Significance Testing. Econometric Theory 16:576–601
5. Pagan AR, Schwert GW (1990) Alternative Models for Conditional Stock Volatility. Journal of Econometrics 45:267–290
6. Petersmeier K (2003) Kerndichte- und Kernregressionsschätzungen im Asset Management. Uhlenbruch, Bad Soden/Ts.

Part XI

Forecasting and Marketing

Detecting and Debugging Erroneous Statements in Pairwise Comparison Matrices

Reinhold Decker, Martin Meißner, and Sören W. Scholz

Business Administration and Economics
Bielefeld University, 33615 Bielefeld, Germany

Summary. Paired comparisons are a widely used technique for evaluating decision elements. If measured on a ratio scale, the judgments form a pairwise comparison matrix that can be interpreted as a ratio preference network. The collection of paired comparisons is often accompanied by manifold response errors driven by variations in attention, mood, mental efficiency, or the general mental state of the respondents. Such erroneous statements might seriously impair the consistency of the responses, and thus, the relative weights derived from the ratio preference network. This paper presents a new error detection approach that identifies deficient elements in pairwise comparison matrices and significantly reduces the mentioned effects with regard to the weights of the considered decision elements. It is based on the geometric mean of all connecting paths of the respective pairwise comparison matrix. Its basic applicability is demonstrated by means of Monte Carlo simulations.

1 Introduction

Paired comparison models address data that arise from situations in which two objects are directly compared to determine the degree of preference. Various approaches have been proposed to investigate the preference structure of respondents including the Bradley-Terry model for paired comparisons based on binary data and ordinal pairwise comparison models for measurements on an ordinal scale (see, e.g. [1]). Moreover, various methods have been proposed to derive preference weights from paired comparisons measured on a ratio scales, most prominently the Eigenvalue method of Saaty's Analytic Hierarchy Process (AHP) [4]. In this paper, we will concentrate on ratio-scaled paired comparison data represented by a matrix $\mathbf{A} = (a_{i,j})_{i,j=1,\ldots,n}$.

Here, $a_{i,j}$ equals the observed ratio for all possible pairs of n decision alternatives and expresses the strength with which decision alternative

i dominates alternative j with respect to a given criterion. Usually, these paired comparisons are collected from consumers' or experts' in marketing research surveys. Matrix **A** specifies a directed graph (as displayed in Figure 1 for $n = 4$) and is also referred to as a ratio preference network [4].

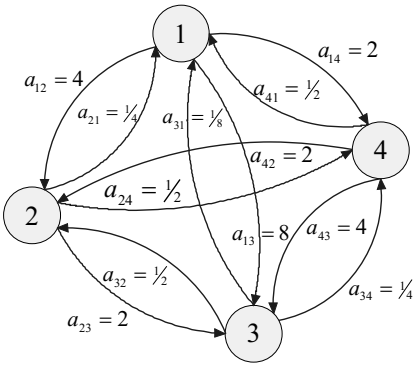


Fig. 1. Ratio preference network with 12 paired comparisons

Ideally, matrix **A** is symmetrically reciprocal. In this case, $a_{i,j} = 1/a_{j,i}$ holds for all entries. Moreover, the ratio preference network is fully consistent if $a_{i,j} = a_{i,k} \cdot a_{k,j}$ holds for all entries (transitivity condition) with $k = 1, \ldots, n$. In the above description of ratio preference networks the transitivity condition can be extended to connected paths with more than two elements:

$$a_{i,j} = a_{i,k_1} \cdot a_{k_1,k_2} \cdot \cdots \cdot a_{k_r,j} \quad \forall i, j \text{ and } k_1, \ldots, k_r \in \{1, \ldots, n\}$$

That is, a pairwise comparison matrix is fully consistent if each connected path results in the same preference ratio (see Figure 1).

In marketing practice, however, the surveyed paired comparisons are rarely fully consistent. Cognitive psychology and human information processing literature teach us that there is considerable noise in the human central nervous system at any given moment influencing the accuracy of preference statements [6]. While small perturbations do not seriously harm the elicitation of the underlying preference structure (see, e.g. [5]), deficient judgments with considerable deviations from the actual preferences of the respondent might seriously impair the resulting preference weights. Therefore, the aim of this paper is twofold. We investigate the structure of erroneous statements in pairwise comparison matrices and propose a new detection approach that identifies deficient elements and significantly reduces the mentioned effects.

In Section 2, we present the concept of consistency in pairwise comparison matrices and discuss sources of erroneous statements in paired comparisons. Section 3 outlines the new error detection approach. In Section 4, the new approach is tested in a Monte Carlo simulation study. The paper concludes with a discussion of the results.

2 Errors in Preference Measurement

Assuming that a respondent has preference structure $\mathbf{w} = (w_1, \ldots, w_n)$ for the n decision alternatives, the above transitivity and symmetric reciprocality conditions hold, if $a_{i,j} = w_i/w_j \; \forall i, j = 1, \ldots, n$. However, some of the decision-maker's statements might deviate from his or her real preference structure when quoting the preferences. The respondent might misstate the direction of his or her real preference structure for instance (i.e. alternative i is preferred to j, but the opposite is stated). We denote this perturbation as error type A. Moreover, the real strength of preference might be over- or underestimated in a stated paired comparison (denoted error type B in the following). Obviously, both errors result in inconsistent ratio preference networks.

We measure violations of the transitivity criterion in order to identify the perturbation of each entry $a_{i,j}$ with respect to the structure of the ratio preference network associated with $\mathbf{A}$. If there exist substantial inconsistencies or erroneous statements in matrix $\mathbf{A}$, then the product of all elements included in one elementary path differs for different elementary paths connecting the elements i and j. However, the aggregation of these estimates provides a robust measure for each entry $a_{i,j}$ which levels out large proportions of errors in the paired comparison data [3]. Therefore, we calculate the estimated average paired comparison value $\overline{a}_{i,j}$ by means of the geometric mean of all elementary paths connecting alternative i and j:

$$\overline{a}_{i,j} = \sqrt[q]{\prod_{g=1}^{q} (a_{i,k_1^g} \cdot a_{k_1^g,k_2^g} \cdot \cdots \cdot a_{k_r^g,j})} \quad \forall k_1^g, \ldots, k_r^g \in \{1, \ldots, n\}$$

A comparison of the stated values $a_{i,j}$ and the average estimated paired comparison values $\overline{a}_{i,j}$ reveals those erroneous elements that do not fit in the ratio preference network. Carré's [2] backtracking algorithm provides an easy means to solve the NP-hard problem of deriving all elementary paths connecting any two elements in the network.

3 Identification of Erroneous Statements in Ratio Preference Networks

To identify those entries $a_{i,j}$ that do not fit in the structure of the ratio preference network, the following three steps have to be conducted:

1. Calculate the deviation between the stated and the estimated average paired comparison $d_{i,j} = |a_{i,j} - \overline{a}_{i,j}|$ for each element of $\mathbf{A}$.
2. Rank the paired comparisons by $d_{i,j}$ in decreasing order.
3. Normalize the deviations by the overall maximum ($d^*_{i,j} = d_{i,j}/d^{max}_{i,j}$).

The resulting deviations $d^*_{i,j}$ are used to examine the structure of the pairwise comparison matrix. Figure 2 visualizes the rationale of our approach for a matrix comprising eight alternatives. Here we assumed that the decision maker gave reciprocal answers (i.e. $a_{i,j} = 1/a_{j,i}$) so that the number of paired comparisons necessary to compare n elements reduces to $n(n-1)/2$. We further assumed that the respondent's judgements are standard normally distributed around his or her true preferences (error type B). Moreover, we changed the direction of some of the paired comparisons to build in errors of type A.

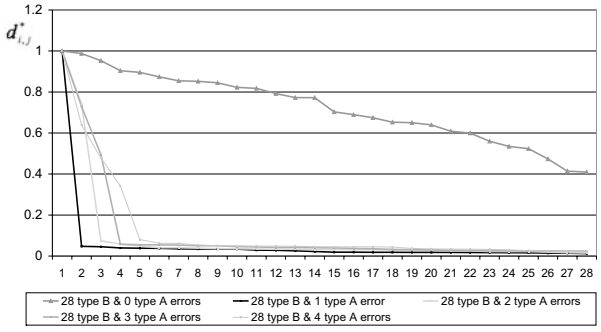


Fig. 2. Visualization of error structures in pairwise comparison matrices by means of ordered deviations $d^*_{i,j}$

Obviously, errors of type A can easily be separated from the considerable noise (error type B) in the paired comparisons. It is easy to see which perturbations result from the ubiquitous noise in human judgments and which elements of the pairwise comparison matrix incorporate a significant deviation from the average estimated values $\overline{a}_{i,j}$. Consequently, the above visualization of the matrix structure provides information about serious errors in the ratio preference network.

To automate the detection of the distorted elements $a_{i,j}$ in matrix $\mathbf{A}$, we make use of the foundations of the theory of (random) measurement

error. Presuming that the measurement errors are standard normally distributed on all entries [8], we suggest to remove those paired comparisons for which $d_{i,j} > \overline{d} + 2\sigma_d$ holds, with $\overline{d}$ denoting the mean and the σ_d standard deviation of $d_{i,j}$. In this way, entries $a_{i,j}$ with deviations lying outside the 95 percent confidence interval are removed and automatically replaced by the average estimated paired comparison values $\overline{a}_{i,j}$. In doing so, the unknown preference structure $\mathbf{w}$ can be derived by standard preference estimation methods. In the following simulation study, we apply Saaty's Eigenvalue method, which is also based on the geometric mean of all paths included in matrix $\mathbf{A}$, and thus, is a good match to our error detection approach.

4 Simulation Study

To illustrate the power of the new error detection approach, we conduct a Monte Carlo simulation study. Since the scale used for preference measurement significantly affects the impact of the above errors, we applied Saaty's 9-point scale, which is the standard in AHP. Thus, the range of the entries $a_{i,j}$ is restricted to the interval $[1/9; 9]$. We constructed 80 fully consistent pairwise comparison matrices of size $n = 8$. Then, all paired comparisons were perturbed by adding values taken from a standard normal distribution (error type B). Scholz et al. [7] have shown that these errors hardly impair the quality of the resulting preference structures. Furthermore, errors of type A have been added to include serious perturbations in matrix $\mathbf{A}$.

To test the effect of these errors on data quality, we compared the MSE between the initial and the distorted preference structures with and without the application of the proposed error detection approach. Table 1 presents the average MSEs for varying numbers of type A and a constant type B error on each entry $a_{i,j}$ of the 80 initial matrices.

Table 1. Average mean squared errors (MSE) for varying numbers of type A errors and type B errors with $\sigma = 1$ for all paired comparisons

Number of type A errors	0	1	2	3	4
MSE without detection	0.0430	0.1186	0.1627	0.1795	0.1914
MSE with detection	0.0520	0.0435	0.0520	0.0669	0.0678

The results show that the perturbations caused by the type B errors are small (see column "0 errors"). However, without error detection, the type A errors cause considerable perturbations in the respective

matrices and the corresponding preference structures. The error detection approach limits the MSE to an almost constant level when type A errors are added. Thus, type A errors are generally correctly identified and eliminated. In all, the new error detection approach enables the robust estimation of the preference structure in case of type A errors.

5 Discussion and Conclusions

This paper presents a new approach to analyze and visualize the structure of perturbations in pairwise comparison matrices. By using a confidence interval, the approach is fully "automated" and needs no further adjustment. It can be run prior to preference estimation and can be combined with any preference elicitation method. Monte Carlo simulations have illustrated the practicability and efficiency of the new approach and showed that in most cases serious errors, such as misstated preference directions, can be predicted and eliminated adequately.

Social and market researchers who deal with paired comparison survey data might benefit from our findings in two ways: First, the visualization of errors helps to understand the quality of the initial data. Second, the automatic error detection approach provides more robust results without further ado. However, further research is necessary to analyze its behavior in case of other error structures.

References

1. Böckenholt U, Dillon WR (1997) Modeling within-subject dependencies in ordinal paired comparison data. Psychometrika 62-3:411–434
2. Carré B (1979) Graphs and Networks. Claredon Press, Oxford
3. Harker PT (1987) Incomplete pairwise comparisons in the Analytic Hierarchy Process. Mathematical Modelling 9-11:837–848
4. Hartvigsen D (2005) Representing the strength and directions of pairwise comparisons. European Journal of Operational Research 163-2:357–369
5. Saaty TL (1980) The Analytic Hierarchy Process. McGraw-Hill, N.Y.
6. Schmidt FL, Hunter JE (1999) Theory testing and measurement error. Intelligence 27-3:138–198
7. Scholz SW, Meißner M, Wagner R (2006) Robust preference measurement: A simulation study of erroneous and ambiguous judgment's impact on AHP and Conjoint Analysis. In: Haasis HD et al. (eds) Operations Research Proceedings 2005. Springer, Berlin
8. Thurstone L (1927) A law of comparative judgment. Psychological Review 34:273–286

Prognose von Geldautomatenumsätzen mit SARIMAX-Modellen: Eine Fallstudie

Stephan Scholze and Ulrich Küsters

Katholische Universität Eichstätt-Ingolstadt, Auf der Schanz 49, 85049 Ingolstadt. {stephan.scholze,ulrich.kuesters}@ku-eichstaett.de

1 Einleitung

Die Optimierung der vorgehaltenen Geldmengen in Geldautomaten sowie deren Bestückungsintervalle erfordert eine genaue Prognose der täglich an Geldautomaten abgehobenen Geldmengen. In dieser Arbeit wird im Rahmen einer Fallstudie der Nutzen konkurrierender Prognoseverfahren verglichen, wobei der Fokus auf SARIMAX-Modellen liegt. Diese berücksichtigen neben saisonalen Effekten auch kausale Kalendereffekte. Zusätzlich werden naive Prognoseverfahren sowie neuronale Netze als Benchmark herangezogen.

Als Datenbasis dieser Fallstudie dient eine Zeitreihendatenbank der Umsätze mehrerer Geldautomaten einer Bank. Exemplarisch wird eine Zeitreihe von 105 Beobachtungen, die den Zeitraum vom 1. Januar bis zum 15. April 2003 abdeckt, ausgewählt. Ziel ist die Konstruktion einer präzisen Prognosefunktion über einen Prognosehorizont von 14 Tagen für jeden Geldautomaten.

Die für diese Fallstudie verwendete Zeitreihe GAA.31 der täglich entnommenen Geldmengen weist ein saisonales Muster auf, das sich vor allem durch geringe Geldentnahmen an Sonntagen bemerkbar macht. Inspiziert man das Saisonmuster mit Hilfe eines Boxplots, so fallen die relativ geringen Schwankungen der Geldentnahmen unter der Woche auf. Lediglich an Donnerstagen und Freitagen nimmt die Geldentnahme geringfügig zu. Darüber hinaus weist die Zeitreihe auch einen schwachen Anstieg der abgehobenen Geldmengen an Monatsenden auf. Die geringe Zeitreihenlänge von drei Monaten ermöglicht aber nur einen eingeschränkten Rückschluss auf systematische Muster.

Problematisch ist der Prognosezeitraum mit einem Horizont von 14 Tagen. Die letzte Beobachtung der Zeitreihe, der 15. April 2003, ent-

spricht dem Dienstag der Karwoche. Damit ist die Modellierung eines zu Ostern zugehörigen Kalendereffektes schwierig. Einerseits liegen keine Informationen aus den Vorjahren über das Geldentnahmeverhalten vor, während und nach den Osterfeiertagen vor. Auf der anderen Seite unterscheiden sich die beiden Feiertage Neujahr (1. Januar) und Heilige Drei Könige (6. Januar) nur geringfügig von einem Sonntag. Dementsprechend schwierig ist die Berücksichtigung eines Ostereffektes. Daher werden die Osterfeiertage einschl. Karfreitag wie Sonntage behandelt.

2 Prognoseverfahren

Aufgrund der oben beschriebenen Effekte werden drei Verfahrensfamilien verwendet:

1. SARIMA(p,d,q)(P,D,Q)-Modelle, mit denen lediglich Saisoneffekte berücksichtigt werden können (saisonale Box-Jenkins Modelle [2]).
2. SARIMAX-Modelle, mit denen zusätzlich auch Regressoren und Interventionen und somit auch Feiertagseffekte berücksichtigt werden können (Transfer- und Interventionsmodelle [9]).
3. Neuronale Netze in Form von Multi-Layer-Perzeptronen, die ausschlieslich als Benchmark verwendet werden [1].

Zur Parameterschätzung werden bei allen Verfahren die ersten 91 Beobachtungen $\{y_1, ..., y_{91}\}$ als Kalibrationsstichprobe verwendet. Die nachfolgenden 14 Beobachtungen $\{y_{92}, ..., y_{105}\}$ dienen als Teststichprobe zur Evaluation der Güte konkurrierender Prognosemodelle. Die erst nach Abschluss der Modellbildung zur Verfügung gestellten Beobachtungen $\{y_{106}, ..., y_{119}\}$ dienen zur Validierung der Ergebnisse.

Saisonale Box-Jenkins Modelle [2] bilden den Datengenerierungsprozess einer saisonalen Zeitreihe y_t durch ein SARIMA(p,d,q)-(P,D,Q)-Modell der Form

$$\phi(B)\Phi(B)\nabla^d\nabla_s^D y_t = \theta(B)\Theta(B)a_t \tag{1}$$

ab. Dabei ist $\phi(B) = 1 - \phi_1 B - ... - \phi_p B^p$ ein autoregressives Polynom der Ordnung p, $\Phi(B) = 1 - \Phi_1 B^s - ... - \Phi_P B^{Ps}$ ein saisonal autoregressives Polynom der Ordnung P, $\theta(B) = 1 + \theta_1 B + \theta_2 B^2 + ... + \theta_q B^q$ ein nicht-saisonales Moving-Average-Polynom der Ordnung q und $\Theta(B) = 1 + \Theta_1 B^s + ... + \Theta_Q B^{Qs}$ ein saisonales MA-Polynom der Ordnung Q ([9], S. 33 und S. 152) mit s als Saisonperiode. Die Differenzoperatoren $\nabla^d = (1-B)^d$ und $\nabla_s^D = (1-B^s)^D$ der Ordnungen d und D eliminieren nichtsaisonale und saisonale stochastische Instationaritäten. Der Innovation-

sterm a_t ist eine Folge stochastisch unabhängiger, identisch $N(0, \sigma^2)$-verteilter Zufallsvariablen. Die Identifikation der Polynom- und Integrationsordnungen p, d, q, P, D und Q kann über visuelle oder semi-automatische Verfahren erfolgen. Zur Festlegung der Integrationsgrade bieten sich zusätzlich saisonale und nicht-saisonale Unit Root Tests an [3]. Spezialfälle des SARIMA(p,d,q)(P,D,Q)-Modells können als Äquivalente oder Approximationen für einige klassische Prognosemodelle verwendet werden. Das SARIMA(0,0,0)(0,1,0)-Modell entspricht einem saisonalen Random Walk. Das SARIMA(0,1,1)(0,1,1)-Modell stellt eine Approximation des Holt-Winters-Modells mit additivem Trend und additiver Saison dar [5].

Ein **Interventionsmodell**

$$y_t = \nu_0 + \sum_{i=1}^{k} \nu_i(B) x_{it} + \frac{\theta(B)\Theta(B)}{\phi(B)\Phi(B)\nabla^d \nabla_s^D} a_t \ , \tag{2}$$

das in der Ökonometrie als SARIMAX-Modell bezeichnet wird, kombiniert dynamische Regressoren mit einem Fehlerterm, der einem SARIMA-Modell (1) folgt. Dabei dient der dynamische Regressionsterm $\nu_0 + \sum_{i=1}^{k} \nu_i(B) x_{it}$ der Beschreibung von Effekten der Variablen x_{it} auf die abhängige Variable y_t. Die direkten, zeitlich verzögerten und vorgelagerten Einflüsse der Variablen x_{it} auf y_t werden durch die Gewichtungspolynome $\nu(B) = \omega(B)B^b/\delta(B)$ abgebildet [9]. In diesem Aufsatz werden als Regressoren nur Kalendereffekte verwendet.

Neuronale Netze in Form von Multilayer-Perzeptronen (MLP) [1]

$$\frac{y_t}{\max_{t=1,\ldots,T}\{y_t\}} = \lambda\left(\alpha + \sum_{h=1}^{H} \omega_h \lambda\left[\alpha_h + \sum_{i=1}^{k} \omega_{hi} x_{it}\right]\right) + a_t \tag{3}$$

für die normierte Variable $y_t^* = y_t / max\{y_t\}$ werden als Benchmark für das SARIMAX-Modell verwendet. Hier ist $\lambda(z) = \exp(z)/(1 + \exp(z))$ eine logistische Aktivierungsfunktion. ω_h für $h = 1, \ldots, H$ stellt die Gewichte der H Knoten der verdeckten Schicht und ω_{hi} die Gewichte der unabhängigen Variablen x_{it} dar. Die Schätzung der Gewichte ω_h und ω_{hi} erfolgt durch das von Venables und Ripley [8] in R [7] implementierte BFGS-Verfahren.

3 Modellselektion

Neben der Nutzung prespezifizierter Modelle, die unter anderem das SARIMA(0,1,1)(0,1,1)-Modell, das SARIMA(0,0,0)(0,1,0) und die Buys-Ballot-Tabelle [9] beinhalten, werden diverse SARIMA-, SARIMAX-

und MLP-Modelle mit Hilfe von automatischen Modellsuchverfahren und Signifikanztests auf Grundlage der Kalibrationsstichprobe ermittelt. Mit Hilfe der Teststichprobe werden anschließend Modelle ausgewählt. Zur Bestimmung der Modellordnungen des SARIMA-Modells wird u.a. das enumerative Modellselektionsverfahren nach Hyndman und Khandakar [4] benutzt. Dieses minimiert das Akaike-Informations-Kriterium AIC [9] und führt zu einem SARIMA(0,0,1)(2,0,0)-Modell.

Optimale SARIMAX-Modelle werden ebenfalls mit einem enumerativen Suchverfahren ermittelt. Anstelle des AIC erfolgt die Auswahl jedoch über den minimalen MAPE [6], der über alle Prognosehorizonte h der Teststichprobe gemittelt wird. Für das Enumerationsverfahren werden folgende Modellparameterbereiche festgelegt:

- SARIMA-Polynomordnungen: $p \in \{0,1,2\}$, $q \in \{0,1,2\}$, $P \in \{0,1,2\}$, $Q \in \{0,1,2\}$, $d \in \{0,1\}$, $D \in \{0,1\}$.
- Exogene Effekte (x_{it}): Monatsende (me), Montag (mo), Dienstag (tu), Donnerstag (th), Freitag (fr), Samstag (sa), Sonntag (su), Ferien (fe).

Die Parameter des Interventionspolynoms $\nu(B) = \omega(B)B^b/\delta(B)$ sind durch $b \in \{0,1,2\}$, $\omega(B) = 1+\omega_1 B+\omega_2 B^2+\omega_3 B^3$ und die Voreinstellung $\delta(B) = 1$ restringiert. Das enumerative Suchverfahren ermittelt das SARIMAX-Modell SM1

$$y_t = \sum_{i \in \{mo,th,sa,su,me\}} \omega_i x_{it} + \frac{(1-\theta_1 B)(1-\Theta_1 B^s)}{(1-\phi_1 B)(1-\Phi_1 B^s)(1-B)} a_t \ . \qquad (4)$$

Im anschließenden Schritt werden insignifikante Parameter aus dem Modell durch ein Abwärtsselektionsverfahren sukzessiv eliminiert (Signifikanzniveau $\alpha = 5\%$). Das zweite SARIMAX-Modell SM2 entsteht nach Entfernen der insignifikanten saisonalen Parameter Φ_1 und Θ_1. Werden in einem dritten Schritt zusätzlich alle insignifikanten Regressoren aus SM2 gestrichen, ergibt sich das SARIMAX-Modell SM3.

Ein ähnliches Suchverfahren wird zur Bestimmung des optimalen MLP implementiert. Zur Abbildung der Eigendynamik des Prozesses werden verzögerte abhängige Variablen spezifiziert: $y^*_{t-1}, y^*_{t-2}, ..., y^*_{t-7}$. Zusätzlich werden 10 unabhängige Variablen zur Einbettung der Saison- und Kalendereffekte als Regressoren eingebettet. Das Selektionsverfahren minimiert ebenfalls den MAPE, der über alle Prognosehorizonte h der Teststichprobe gemittelt wird. Ergebnis ist ein neuronales Netz mit den um zwei und drei Perioden verzögert normierten Inputvariablen y^*_{t-2} und y^*_{t-3}, den kontemporären Variablen me_t, sa_t und su_t sowie der um zwei Perioden nach vorne verschobenen Lead-Variablen me_{t+2}.

Table 1. Vergleich der Prognosegüte durch MAPE und RMSE für die einzelnen Teilstichproben $\{y_1, ..., y_{91}\}$, $\{y_{92}, ..., y_{105}\}$ und $\{y_{106}, ..., y_{119}\}$

Modell	$\{y_1, ..., y_{91}\}$		$\{y_{92}, ..., y_{105}\}$		$\{y_{106}, ..., y_{119}\}$	
	MAPE	RMSE	MAPE	RMSE	MAPE	RMSE
SARIMA(0,0,1)(2,0,0)	71.01%	16234	10.55%	6321	43.40%	26519
SARIMA(0,0,0)(0,1,0)	73.59%	20657	14.74%	18902	97.90%	26065
SARIMA(0,1,1)(0,1,1)	50.18%	13508	21.72%	11751	114.73%	29118
Buyes-Ballot	52.55%	14007	7.36%	6552	38.83%	12764
SM1	45.90%	13407	*2.22%*	*2287*	*13.46%*	7510
SM2	*45.75%*	13853	5.33%	4028	13.58%	*7404*
SM3	46.28%	14181	6.09%	4756	23.48%	9243
MLP	59.51%	*11067*	3.01%	3196	14.56%	9437

4 Ergebnisse und Prognosen

Die Prognosegüte wird durch den MAPE und den RMSE gemessen. Dabei werden einerseits die einstufigen Prognosefehler für die Kalibrationsstichprobe $\{y_1, ..., y_{91}\}$, andererseits die über den Prognosehorizont H kumulierten Fehlermaße für die Teststichprobe $\{y_{92}, ..., y_{105}\}$ und die außerhalb der Schätzbasis liegende Kontrollstichprobe $\{y_{106}, ..., y_{119}\}$ inspiziert. Das Ergebnis in Tabelle 1 zeigt deutlich, dass die SARIMAX-Modelle sowohl prespezifizierten als auch automatisch bestimmten SARIMA- und MLP-Modellen überlegen sind. Gemessen am MAPE schneiden die geringfügig überspezifizierten SARIMAX-Modelle SM1 und SM2 am besten ab. Den geringsten RMSE für den einstufigen expost Prognosefehler weist das MLP-Modell aus. Dieser geringe Anpasssungsfehler des MLP in der Kalibrationsstichprobe lässt sich aber nicht auf die Prognosegüte in der Test- und Kontrollstichprobe übertragen. Dies deutet auf eine Überparametrisierung des MLP-Modells hin. In der Kontrollstichprobe weist das Interventionsmodell SM2 den geringsten RMSE auf.

Im SM2-Modell wurden nur die insignifikanten saisonalen SARIMA-Parameter des SM1-Modells, nicht aber die Kalendereffekte eliminiert. Dies impliziert, dass eine leichte Überspezifikation exogener Saison- und Kalendereffekte auch bei insignifikanten Parameterschätzern nützlich ist. Bei den anderen Zeitreihen ergeben sich ähnliche Ergebnisse. Abbildung 1 zeigt die Prognose des SM1-Modells einschließlich der 95%-igen Konfidenzintervalle.

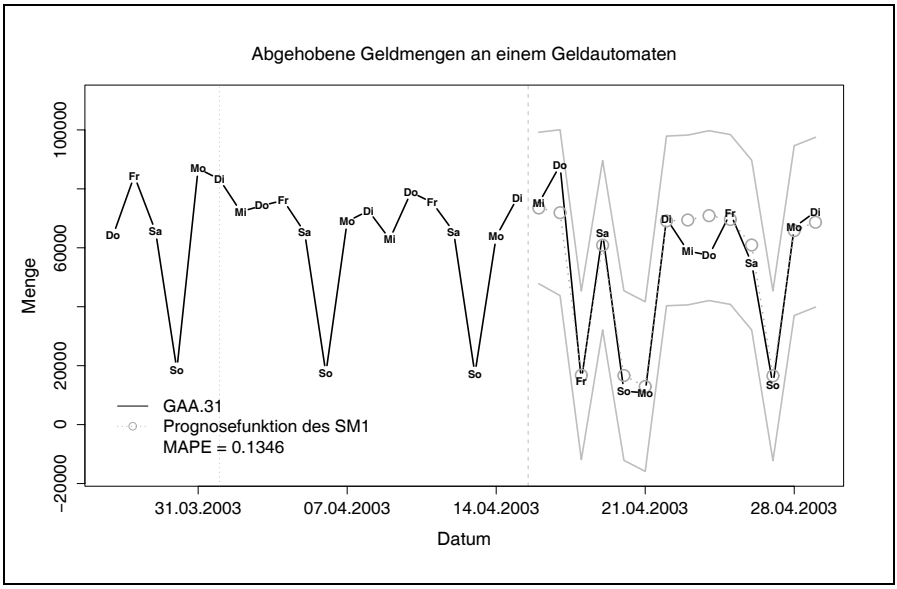


Fig. 1. Prognose der Beispielzeitreihe GAA.31 mit einem SARIMA(1,1,1)(1,0,1)-Modell und den zusätzlichen exogenen Effekten Montag, Donnerstag, Samstag, Sonntag sowie Monatsende für den Prognoseursprung t = 106 und den Prognosehorizonten $h = 1, ..., H = 14$

References

1. Bishop, CM (1995) Neural Networks for Pattern Recognition. Oxford University Press
2. Box GEP, Jenkins GM, Reinsel GC (1994) Time Series Analysis, Forecasting and Control. Prenctice Hall
3. Ghysels E, Osborn DR (2001) The Econometric Analysis of Seasonal Time Series. Cambridge University Press
4. Hyndman RJ, Khandakar Y (2007) Automatic Time Series Forecasting: The Forecast Package for R. Journal of Statistical Software, to appear
5. Kendall MG, Ord K (1989) Time Series. 3. Auflage, Arnold
6. Küsters U (2005) Evaluation von Prognoseverfahren. In: Mertens P, Rässler S (Hrsg.) Prognoserechnung. 5. Auflage, Physica Verlag
7. R Development Core Team (2007) R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org.
8. Venables WN, Ripley BD (2002) Modern Applied Statistics with S. 4. Auflage, Springer
9. Wei WWS (2006) Time Series Analysis: Univariate and Multivariate Methods. 2. Auflage, Pearson Addison Wesley

Part XII

Health Care Management

On Dimensioning Intensive Care Units

Nico van Dijk and Nikky Kortbeek

Operations Research Group, Department of Economics and Business,
University of Amsterdam, Roetersstraat 11, 1018 WB Amsterdam,
The Netherlands. n.m.vandijk@uva.nl, n.kortbeek@uva.nl

Summary. Due to a limited ICU capacity patients can be rejected at both the Operating Theater (OT) and at the Intensive Care Unit (ICU) within hospitals. The corresponding ICU-rejection probability is an important service factor for hospitals. A simple expression for this probability is not available. With c the ICU capacity (number of ICU beds), this paper provides analytic support for:

(i) An $M|G|c|c$-approximation.
(ii) A secure $M|G|c\text{-}1|c\text{-}1$ upper bound.

The upper bound can be of practical interest so as to dimension the size of an ICU to secure a sufficiently small rejection probability.

1 Introduction

1.1 Motivation

At an ICU patients are admitted for intensive care, such as monitoring and artificial ventilation. Patients may also require an ICU bed for postoperative care after a heavy operation. Unfortunately, due to the limited number of beds, a request for an ICU bed may be rejected.

For patients a rejection may lead to further delay in a critical situation which may even put lives at risk. For the hospital (or public health) a rejection may lead to an idle operating room, which is regarded as a loss of precious capacity. The size of an ICU thus needs to be dimensioned carefully.

A careful estimation of the ICU rejection probability is thus required. Unfortunately, measurements might not be available or be sufficiently predictive for different number of beds. An analytic or numeric approach would therefore be of practical interest.

1.2 Literature and Objectives

By a number of references the standard $M|G|c|\cdot$ multi-server queue has already been argued as a reasonable approximate for the ICU in isolation (see [1] and references therein). Nevertheless, these results do not contain:

- A formal justification.
- The inclusion of the OT and its interaction with the ICU.
- A secure lower and upper bound for the ICU-rejection probability.

These are the main objectives of this paper.

2 Original Model Formulation

2.1 Patient Types and Case Study

The inflow of the ICU consists of emergency patients (the majority) and elective patients and can be subdivided into various patient groups. However, as we are particularly interested in the effect of the limited ICU capacity and its interaction with the OT, in this paper we only make a cross distinction in patients, that need to visit the ICU after having undergone an operation, and patients that enter the ICU directly without operation. These patients will be referred to as:

- OT (or type 1-) patients.
- Direct (or type 2-) patients.

This distinction is made:

- To capture the interaction between OT and ICU.
- As the average sojourn times at the ICU significantly differ.

A Case Study

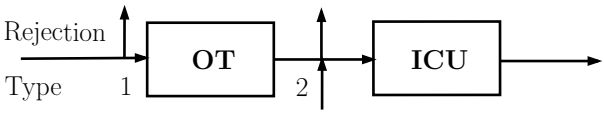
Data were collected for a case study over a one year period. The percentages of type 1 and type 2-patients were 39% and 61%.

The average sojourn time spend in the ICU over all patients was 5.2 days, for roughly 4 days for type 1-patients and 6 days for type 2-patients. Other important characteristics were:

- OT capacity (number of operating rooms): 8.
- ICU capacity (number of beds): 12.
- ICU occupancy: 85%.

2.2 Original Model

To study the ICU-rejection probability $\boldsymbol{R}$ and its interaction with the OT a number of assumptions are made. In [2] each of these assumptions has been argued and justified by simulation to be quite reasonable for practical modeling. The corresponding tandem queue system under these assumptions (1)-(8) will be referred to as the *original* OT-ICU model, and will be studied in sections 4 and 5 as our system of interest.



(1) Patients that do not require an ICU bed are not included.
(2) A Poisson arrival rate λ_1 of OT-patients (type 1) at the OT.
(3) A Poisson arrival rate λ_2 of Direct patients (type 2) at the ICU.
(4) An exponential service time for the surgery at the OT with rate μ_1.
(5) A (possibly non-exonential) sojourn time at the ICU with mean τ_1 for OT-patients and τ_2 for Direct patients.
(6) The OT has c_1 identical operating rooms with a infinite waiting facility; The ICU has a limited capacity for at most c_2 patients and no waiting facility.
(7) When no ICU bed is available, type 1-patients are rejected upon arrival at the OT and type 2-patients are rejected upon arrival at the ICU.
(8) An ongoing operation is always continued. When no ICU bed is available, the patient is kept in the recovery.

3 A Modified OT-ICU System

The original OT-ICU system of interest has no analytic solution. However, in line with literature (e.g. [1]), the ICU-rejection probability seems to be approximated reasonably well by Erlang's loss expression (6); more precisely that is, by an $M|G|c|c$-queue with $c = c_2$ the number of ICU beds. This section provides formal support for this approximation as based upon the following artificial modification of (8):

(8') When the ICU becomes congested, operations are immediately interrupted and stopped. The operations are resumed as soon as the ICU is no longer congested.

Under this modification, the tandem system will be referred to as the *modified* OT-ICU system. For this system the following result can be proven.

Theorem 1. *Let $(n_1; m_1, m_2)$ denote that there are n_1 patients at the OT and m_i patients at the ICU of type i $(i = 1, 2)$. For the modified OT-ICU system, with $m = m_1 + m_2 \leq c_2$,*

$$F_1(n_1) = [n_1!]^{-1} \ \ for \ \ n_1 \leq c_1 \ \ and \ \ [c_1! c_1^{(n_1 - c_1)}]^{-1} \ \ for \ \ n_1 > c_1 \quad (1)$$

and with normalizing constant α, we have:

$$\pi(n_1; m_1, m_2) = \alpha \, F_1(n_1) \left(\frac{\lambda_1}{\mu_1}\right)^{n_1} \frac{1}{m_1!} (\lambda_1 \tau_1)^{m_1} \frac{1}{m_2!} (\lambda_2 \tau_2)^{m_2} \quad (2)$$

Proof. For selfcontainedness of this paper we restrict to a compact proof for the exponential case with exponential parameters $\gamma_t = 1/\tau_t$, $t = 1, 2$ at the ICU. The proof for the general non-exponential case can be found in [2]. For $t = 1, 2$, let

$$\mu_1(n_1; m_1, m_2) = \begin{cases} n_1 \mu_1 & , m_1 + m_2 < c_2 \text{ and } n_1 < c_1 \\ c_1 \mu_1 & , m_1 + m_2 < c_2 \text{ and } n_1 \geq c_1 \\ 0 & , m_1 + m_2 = c_2 \end{cases} \quad (3)$$

$$\mu_{2t}(m_t) = m_t \gamma_t \quad , \quad \lambda_t(m_1, m_2) = \begin{cases} \lambda_t & , m_1 + m_2 < c_2 \\ 0 & , m_1 + m_2 = c_2 \end{cases} \quad (4)$$

We need to verify the global balance equation for any state $(n_1; m_1, m_2)$ to equate the total outrate (the left hand side) and the total inrate (the right hand side). It will be convenient to order the detailed outrates and inrates as:

$$\begin{cases} \pi(n_1; m_1, m_2)\mu_1(n_1; m_1, m_2) + & (5.1) \\ \pi(n_1; m_1, m_2)\mu_{21}(m_1) + & (5.2) \\ \pi(n_1; m_1, m_2)\mu_{22}(m_2) + & (5.3) \\ \pi(n_1; m_1, m_2)\lambda_1(m_1, m_2) + & (5.4) \\ \pi(n_1; m_1, m_2)\lambda_2(m_1, m_2) & (5.5) \end{cases}$$

$$= \quad (5)$$

$$\begin{cases} \pi(n_1 - 1; m_1, m_2)\lambda_1(m_1, m_2) 1_{(n_1 > 0)} + & (5.1)' \\ \pi(n_1 + 1; m_1 - 1, m_2)\mu_1(n_1 + 1, m_1 - 1, m_2) 1_{(m_1 > 0)} + & (5.2)' \\ \pi(n_1; m_1, m_2 - 1)\lambda_2(m_1, m_2 - 1) 1_{(m_2 > 0)} + & (5.3)' \\ \pi(n_1; m_1 + 1, m_2)\mu_{21}(m_1 + 1) 1_{(m_1 + m_2 < c_2)} + & (5.4)' \\ \pi(n_1; m_1, m_2 + 1)\mu_{22}(m_2 + 1) 1_{(m_1 + m_2 < c_2)} & (5.5)' \end{cases}$$

This global balance equation (5) is ordered as if it can be decomposed into five local balances. The proof can be completed directly by substituting (2) which equates each of the detailed equations $(5.i)$=$(5.i)'$, $i = 1, \ldots, 5$. $\square$

Remark 1 (Literature). Even for the exponential case the product form result (2) can be regarded as 'new' as it combines

- a non-reversible routing with blocking and
- multiple job types.

Remark 2 (Erlang loss expression). The product form expression (2) decomposes as if the OT and ICU can be regarded as indepenent. As a consequence, theorem 1 directly justifies an $M|G|c|c$-loss approximation for the ICU rejection probability with $c = c_2$. More precisely, straightforward arranging terms of (2) for $\pi(m_1 + m_2 = c_2)$ yields:

Corollary 1. *For the modified OT-ICU system with $m = m_1 + m_2 \leq c_2$ and arbitrary nonnegative ICU-sojourn time distributions for OT (type 1-) and Direct (type 2-) patients, the ICU-rejection probability for type 1-(at the OT) and type 2-(at the ICU) patients is determined by the loss expression:*

$$\boldsymbol{B}(c) = \rho^c/c! \left[\sum_{k=0}^{c} \rho^k/k!\right]^{-1} \quad with \;\; \rho = (\lambda_1\tau_1 + \lambda_2\tau_2) \tag{6}$$

4 Bounds

Intuitively, as the modified OT-ICU tandem system only differs from the original OT-ICU tandem system for a patient in operation when the ICU becomes congested, one may expect that the $M|G|c|c$-loss expression, as based upon corollary 1, is a quite reasonable if not accurate approximation for the original OT-ICU system. This appears to be true (as has already been used in the literature, but without formal justification). In fact, by using the modified system for $c_2 = c$ and $c_2 = c - 1$, and result 1, the following main result can be proven, which provides secure bounds.

Theorem 2. *With*

- $\boldsymbol{R}$ *the ICU-rejection probability upon arrival at the OT for a type 1-patient and at the ICU for a type 2-patient for the original OT-ICU system and*
- $\boldsymbol{B}(c)$ *the loss probability of an Erlang loss system with c servers with arrival rate $\lambda = \lambda_1 + \lambda_2$ and mean service time τ as in (6):*

$$\boldsymbol{B}(c) \leq \;\; \boldsymbol{R} \;\; \leq \boldsymbol{B}(c-1) \tag{7}$$

Proof. Despite intuition, a proof as based upon a sample path comparison approach can be expected to be highly complicated and have as yet not been established. A technical analytical proof can be found in [2] as based upon a Markov reward proof technique. □

5 Application: Case Study

The case study situation is within the range of realistic figures as recently reported by the Dutch ministry of health. It reports that roughly 10% of ICU requests are strictly rejected, 3% admitted by a predischarge and 4% placed differently. Furthermore an occupancy of 75% is mentioned as norm.

Simulation results for the case study consistently support the lower and upper bound. Particularly, for smaller rejection probabilities, say in the order of 5-10% as for larger hospitals with a high occupancy level, the bounds appear to be quite accurate (in absolute sense). The results seem useful, at least, for practical purposes such as to guarantee a sufficiently small rejection percentage by the upper bound.

For the case study, an occupancy of 85% and 12 beds were used. The results lead to a lower bound of .127 and an upper bound of .172 (the simulation result was .128). As a direct application of the secure $M|G|c\text{-}1|c\text{-}1$ upper bound computation the required number of ICU beds could be computed to guarantee a specified rejection probability $\boldsymbol{R}$, such as:

- 16 beds for at most 5%.
- 19 beds for at most 1%.

6 Conclusion

The rejection probability for an ICU bed is of considerable interest within hospitals. This paper provided support in a twofold manner:

(i) In a practical way, by an easily computable approximation and a secure upper bound for the ICU-rejection probability.

(ii) In a theoretical way, by an analytic justification of this approximation and bound.

As such the paper illustrates how Operations Research can provide both practical and formal support for decision making in health care.

References

1. Litvak, N., Rijsbergen, M., Boucherie, R.J. and Houdenhoven, M. (2006). Managing the overflow of intensive care patients. *European Journal of Operational Research* **185** (2008) 998-1010.
2. Van Dijk, N.M. and Kortbeek, N. (2007). Approximation and bounds for the ICU-rejection probability in OT-ICU tandem systems. *Research Report, Dept. of Economics and Business University of Amsterdam.*

A Hybrid Approach to Solve the Periodic Home Health Care Problem

Jörg Steeg and Michael Schröder

Fraunhofer Institut für Wirtschafts- und Technomathematik,
Fraunhofer-Platz 1, 67663 Kaiserslautern, Germany
joerg.steeg@itwm.fraunhofer.de

Summary. Home health care (HHC) services provide nursing assistance to the elderly with the advantage of allowing them to continue living at their homes. Usually a HHC service has a fleet of vehicles that are used by nurses to get to the patients, where they have to perform a specified job.

The HHC problem consists of the following task: assign a nurse to each job such that several conflicting objective functions are minimized, while a number of constraints are met. From a mathematical point of view, this problem is hard to solve, since it combines two well-known NP-hard problems: the vehicle routing problem and the nurse rostering problem. Our main objective is to minimize the number of nurses that visit a patient during the schedule: the nurse-patient loyalty. We consider this loyalty as a main indicator for a good schedule. Additionally, we have a periodic model, i.e. we do not only want to assign the nurses for one day, but we usually want to plan for a whole scheduling horizon, typically a week. As a solution approach, we propose a hybrid approach that combines the strengths of constraint programming and the large neighborhood search metaheuristic.

1 Introduction

Home Health Care (HHC) services are becoming increasingly important. Their patients receive medical treatments at home. Therefore a HHC service provider has a fleet of vehicles used by nurses to travel to the patients. Usually, there are various shifts that can differ in the set of nurses available and jobs required. A periodic aspect is also involved, since the jobs need to be performed repeatedly during the schedule. In practice, the HHC problem is solved manually by a senior nurse, who often spends a whole day creating next week's schedule.

From a mathematical point of view, this problem is of special interest, because it involves two NP-hard subproblems: the nurse rostering

problem (NRP) and the vehicle routing problem (VRP). The former arises from the fact that each nurse is only available for some of the shifts. Furthermore, their working times are usually restricted to a maximum number of hours a week. The VRP is also part of the planning problem, since travel distances should be minimized. It is periodic and has time windows. The time windows arise from the patients' preference to be treated at certain times. Hence, the HHC problem encapsulates a periodic vehicle routing problem with time windows (PVRPTW).

Only limited research has been conducted into the Home Health Care problem. It seems to have been first described by Cheng and Rich [2]. In their paper, they tackle the problem with two MIP formulations and a two-phase construction heuristic. A more recent article by Bertels and Fahle [1] combines constraint programming (CP) with meta-heuristics to solve the HHC problem. Finally, a paper by Eveborn, Flisberg, and Rönnqvist [4] models the Home Health Care problem for a single day as a set partitioning problem.

This paper is organized as follows: section 2 presents our model, while section 3 presents the hybrid constraint programming – large neighborhood search approach. We conclude by presenting first computational results.

2 A Model for Home Health Care Planning

In this section, we present our new model for the Home Health Care problem, which combines a Nurse Rostering problem with a Periodic Vehicle Routing problem. Usually, the literature considers the problems separately, or Home Health Care problems with a single shift only.

In general, the problem consists mainly of two elements: jobs and nurses. The task is to assign to each job a nurse such that the assignment is feasible. This takes place in a planning horizon consisting of S shifts. Each shift has a time window $[0..H]$.

A job in the planning horizon represents a task at a patient's home. Different jobs can affect the same patient. A job j is characterized by a hard time window $[hbs_j..hbe_j]$ and a processing time pt_j. Furthermore, it has a frequency f_j, which states how many times the job must be processed during the planning horizon. The shifts in which the job has to be performed are given by a set of possible shift combinations $R_j = \{R_{j,1}, \ldots, R_{j,K_j}\}$. The sets $R_{j,l}$ are subsets of shifts, and it yields $|R_{j,l}| = f_j \ \forall l$. Finally we have a distance matrix $D = (d_{ij})$ that states the driving distance between two jobs i and j.

For each nurse n, we have an *a priori* availability $a_n^s \; \forall s$ with $a_n^s = 1$, if nurse n is available for shift s, 0 otherwise. If a nurse is unavailable in this sense, she cannot be made available – not even by paying a penalty cost. Furthermore, a nurse has a designated working time wt_n for the whole schedule. If she is assigned to a shift, the whole length of the shift is added to her working time. Any work of a nurse in addition to her designated working time is penalized by costs c_n per time unit.

A solution to the Home Health Care problem is evaluated by the following criteria:

1. *Nurse-patient loyalty*: We believe that it is important for the patient's satisfaction that he does not have to deal with many different nurses. Advantages are twofold: on the patient's side, it is easier to develop a close relation to the nurse, while on the nurse's side, time can be saved. This is for example because the nurse knows her way around in the patient's household.
2. *Nurse costs*: Usually, a nurse's contract includes a number of hours, e.g., per week, that are covered by her salary. Additional work must be paid with an overtime cost per extra time unit. As soon as a nurse performs at least one job in a shift, we take the whole shift as working time into account.
3. *Traveling distance*: Lastly, we seek to minimize the total travel distance.

We combine these three objective functions with a weighted sum. Then, the optimal schedule is found, if the following term is minimized:

$$\alpha_1 \underbrace{\sum_{j=1}^{J} (|X_j| - 1)}_{\text{criterion 1}} + \alpha_2 \underbrace{\sum_{n=1}^{N} c_n \Big(\max \big\{ 0, H \sum_{s=1}^{S} Y_n^s - wt_n \big\} \Big)}_{\text{criterion 2}} + \alpha_3 \underbrace{\sum_{s=1}^{S} \sum_{n=1}^{N} T_n^s}_{\text{criterion 3}},$$

where $|X_j|$ is the number of nurses used for job j throughout the schedule, and T_n^s is the distance traveled by nurse n in shift s. The positive weights α_i scale the criteria such that they are comparable and state their relative importance.

3 Hybrid Approach

As mentioned before, our model consists of two distinct problems, which have different solution approaches. For the nurse rostering problem, constraint programming (CP) is successful (see [6]), while for the vehicle routing problem metaheuristics are a good choice (see [3]). To tackle

the HHC, we combine the strengths of both approaches. First, we apply a CP goal to compute an initial feasible solution to the problem. Then, a metaheuristic tries to improve the solution, while an underlying constraint programming layer assures feasibility with the nurse rostering problem. In this way, we find a feasible solution quickly with CP, while the metaheuristic lets us navigate through the search space towards optimal solutions. In the rest of this section, we describe the constraint programming approach and the metaheuristic in turn.

The initial solution is found with a constraint programming goal and a guided branch & bound search. In this search, we first fix the shift combinations and then by increasing earliest start times assign the jobs to nurses.

As a metaheuristic for the vehicle routing problem, we chose the recently developed *Adaptive Large Neighborhood Search (ALNS)* (see [5]). It follows a different approach to local search techniques. Instead of changing a solution only slightly, the solution is changed significantly. To perform moves from one solution to another, *remove* and *insert operations* are defined. A remove operation determines the parts of the solution that are allowed to be changed, while the insert operation decides how the removed parts are inserted again, such that we obtain a new feasible solution. For each move, the remove and insert operation are selected randomly. By using weights, the search becomes adaptive. In the beginning, each remove and insert operation has the same weight, i. e. a probability to be chosen. During the search, the weight is adapted by a score that depends on the success of a move. The ALNS is guided by a local search, in our case a Simulated Annealing metaheuristic.

A remove operation is used to identify those jobs which can be shifted to achieve a new solution. These jobs are stored in a request bank. In our current approach, we used the following three remove operations: In *Random removal*, q jobs are randomly removed. In *Worst removal*, we seek to remove the jobs that have the highest cost. Therefore, we sort the routes of the nurses according to the number of jobs they include. Then, we remove routes, starting from the one with the least jobs in it, until we have removed at least q jobs. The reasoning behind this approach is that a job that uses only a single nurse in a shift causes a high cost, since the entire shift is opened for it. Finally, in *Shift combination removal*, we change the shift combinations of q jobs.

An insert operation is a construction heuristic to find a new feasible solution. It is time consuming and difficult to construct a new solution, since many constraints have to be considered. The following insertion operations are available: The *In Order insertion* tries all possible inser-

tions in routes of nurses for the first given job. Then the job is inserted at the position into the route that creates the smallest objective function. Afterwards this procedure is repeated for the second job in the request bank and so on. The *Greedy insertion* determines the inserting position causing a minimal objective function for all jobs in the request bank. Then the job with the minimal cost is inserted. Afterwards, for all remaining jobs, the best insertion position is recomputed and the best job is inserted. This procedure is repeated until no more jobs remain in the request bank.

4 Computational Results

We evaluated the performance of our algorithms with some randomly generated data. Since to our knowledge no standard benchmark instances exist for the Home Health Care problem, we took instances for the Periodic Vehicle Routing Problem with Time Windows (PVRPTW) created by Cordeau (see [3]) and extended them for our problem with a nurse rostering part. Therefore, we made the following design decisions: For each nurse n, we assumed that she is designated to work half of the shifts. Hence, $wt_n = \left\lceil \frac{\#\text{ shifts}}{2} \right\rceil \cdot 300$, where 300 is the general shift length. For each overtime unit, a cost of 1 was assumed ($c_n = 1\ \forall n$). Finally, a nurse is available in a shift with probability 0.75.

For parameter q, the number of jobs to be removed in a move, we followed the suggestions of [5]. Therefore q is randomly drawn from the interval $[a, b]$, where $a = \min\{0.1 \cdot J, 30\}$, $b = \min\{0.4 \cdot J, 60\}$, and J is the total number of jobs.

The results of this computational example are given in table 1. It reads as follows: First, we generated for each instance an initial solution with the Constraint Programming goal. Then, the adaptive large neighborhood search tried to improve the solution until either 1000 moves were performed, or a time limit of 120 minutes was reached. The results show that the LNS could improve the solution significantly in 50% of the instances. For the other instances, only a few LNS moves could be performed, and hence the solution was only improved slightly (or not at all). We believe that these initial computational results prove that the adaptive large neighborhood search is a successful approach to solve the Home Health Care problem if a significant number of moves can be computed.

Table 1. Computational results for the tests with the Cordeau instances

	Inst. Para- meter			Initial	Comp.	LNS	Result	
Instanz	Nurses	Jobs	Tasks	solution	Time	Best sol.	Improv.	Min
pr01	6	48	96	0.6893	0.33 s	0.3709	46.2%	36
pr02	12	96	192	0.8757	0.63 s	0.5030	42.6%	120
pr03	18	144	288	0.9358	2.44 s	0.7169	23.4%	120
pr04	24	192	384	1.0268	8.98 s	0.8270	19.5%	120
pr05	30	240	480	0.9426	21.94 s	0.9145	3.0%	120
pr06	36	288	576	1.0563	44.45 s	1.0466	0.9%	120
pr07	10	72	216	1.3281	0.58 s	0.9882	25.6%	114
pr08	20	144	432	1.4847	8.05 s	1.2642	14.9%	120
pr09	30	216	648	1.6167	40.33 s	1.5915	1.6%	120
pr10	40	288	864	1.7102	119.88 s	1.7102	0.0%	120
pr11	6	48	96	0.8400	0.34 s	0.3330	60.4%	37
pr12	12	96	192	0.9448	0.63 s	0.4379	53.6%	120
pr13	18	144	288	0.9448	2.53 s	0.6322	33.1%	120
pr14	24	192	384	0.9606	9.00 s	0.8797	8.4%	120
pr15	30	240	480	0.9622	21.86 s	0.9622	0.0%	120
pr16	36	288	576	0.9854	44.49 s	0.9854	0.0%	120
pr17	8	72		n.a.	n.a.			120
pr18	16	144	432	1.5509	5.25 s	1.2357	20.3%	120
pr19	24	216	648	1.4984	27.03 s	1.4376	4.1%	120
pr20	32	288	864	1.6817	80.00 s	1.6731	0.5%	120

References

1. S. Bertels and T. Fahle. A hybrid setup for a hybrid scenario: combining heuristics for the home health care problem. *Computers & Operations Research*, 33:2866–2890, 2006.
2. E. Cheng and J. L. Rich. A Home Health Care Routing and Scheduling Problem. Technical Report TR98-04, Department of CAAM, Rice University, USA, 1998.
3. J.-F. Cordeau, M. Gendreau, and G. Laporte. A tabu search heuristic for periodic and multi-depot vehicle routing problems. *Networks*, 30:105–119, 1997.
4. P. Eveborn, P. Flisberg, and M. Rönnqvist. Laps Care– an operational system for staff planning of home care. *European Journal of Operational Research*, 171:962–976, 2006.
5. S. Ropke and D. Pisinger. A general heuristic for vehicle routing problems. *Computers & Operations Research*, 34(8):2403–2435, August 2007.
6. L.-M. Rousseau, M. Gendreau, and G. Pesant. A General Approach to the Physician Rostering Problem. *Annals of Operations Research*, 115:193–205, September 2002.

Tactical Operating Theatre Scheduling: Efficient Appointment Assignment

Rafael Velásquez[1], Teresa Melo[1,2], and Karl-Heinz Küfer[1]

[1] Fraunhofer Institute for Industrial Mathematics (ITWM), D 67663 Kaiserslautern, Germany
{rafael.velasquez,karl-heinz.kuefer}@itwm.fraunhofer.de
[2] University of Applied Sciences, D 66123 Saarbrücken, Germany
teresa.melo@htw-saarland.de

Summary. Finding an appointment for elective surgeries in hospitals is a task that has a direct impact on the optimization potential for offline and online daily surgery scheduling. A novel approach based on bin packing which takes into account limited resource availability (e.g. staff, equipment), its utilization, clinical priority, hospital bed distribution and surgery difficulty is proposed for this planning level. A solution procedure is presented that explores the specific structure of the model to find appointments for elective surgeries in real time. Tests performed with randomly generated data motivated by a mid size hospital suggest that the new approach yields high quality solutions.

1 Introduction and Problem Description

Scheduling elective surgeries at a tactical level deals with finding an appointment for a surgery over a planning horizon of several weeks, while booking individual or generic resources for the particular appointment day. Its output is used as input for the operational planning level problem, which consists in assigning an estimated start and end time to each surgery, as well as solving the corresponding rostering problem.

The importance of an efficient management of operating theatre (OT) resources has been widely documented (refer e.g. to [1]). An important part of a hospital's budget is spent in the OT. Guaranteeing a reduction in idle times of equipment and operating rooms, which in turn results in a more efficient use of the staff's available time, is motivation enough to study this problem in depth.

In practice, the tactical operating theatre scheduling problem (TOTSP) is hardly ever considered as surgery appointments are given on a "first come, first served" basis. The goal of this paper is to present a mathematical model which provides a decision support for finding not only an available and feasible appointment, but also one that will achieve desired criteria. In this sense, the TOTSP will be modelled as a multi-dimensional online packing problem with time windows. Sect. 2 describes the mathematical formulation for the TOTSP. Sect. 3 presents the exact solution method. Sect. 4 reports the computational experience and compares the performance of the proposed approach with the "first fit" (FF) method often used in practice. Finally, Sect. 5 presents some conclusions and directions for further research.

2 Mathematical Formulation of the TOTSP

The problem of finding an appointment for a surgery is considered over a desired time window which describes the patients' preferences for the surgery day and the medical requirements predefined by the surgeon. It is assumed that all departments of the hospital have information regarding resource availabilities and that staff qualifications have been identified. The required notation is introduced as follows:

Index sets T = Set of days resulting from the discretization of the planning horizon (e.g. a year is divided into 365 days); S = Set of elective surgeries, of which the first s_{i-1} already have a fixed appointment and surgery s_i is the surgery that needs to be assigned to $t \in T$; R = Set of resources (e.g. rooms, equipment, staff, beds); R_s = Set of resources required to carry out surgery s; $I = [\underline{I}, \overline{I}]$ = Set of days during which surgery s_i is desired to take place ($I \subset T$).

Parameters c_1, c_2 = penalty weights awarded to scheduling a surgery with overtime or outside the time window I, respectively; ℓ_{rs_k} = amount of time required of resource r for surgery s_k; u_{rs_k} = number of days in which surgery s_k requires resource r; q_{rt} = regular capacity of resource r on day t; v_{rt} = additional capacity of resource r on day t; $M = \max_{r \in R, t \in T}\{q_{rt}\}$.

Decision variables $x_{s_i t} = 1$ if surgery s_i is assigned to day t, and 0 otherwise; $y_{rt} = 1$ if resource r does not require additional capacity on day t, and 0 otherwise.

The following packing formulation is proposed to model the TOTSP, given that surgery s_i has a desired time window $[\underline{I}, \overline{I}]$ during which the surgery should be scheduled and that all previous surgeries $s_1, \ldots, s_{i-1}$ already have a fixed appointment.

$$\text{Min} \quad c_1 \sum_{t \in T \setminus I} x_{s_i t} \; + \; c_2 \; [\, 1 - (\, \sum_{t \in T} (x_{s_i t} \cdot \prod_{r \in R_{s_i}} y_{rt}) \,) \,] \tag{1}$$

$$\text{subject to} \qquad \sum_{t=\underline{I}}^{|T|} x_{s_i t} = 1 \tag{2}$$

$$\sum_{t=1}^{\underline{I}-1} x_{s_i t} = 0 \tag{3}$$

$$\sum_{k=1}^{i} \ell_{r s_k} [x_{s_k t} + \sum_{j=2}^{u_{r s_k}} x_{s_k t-j+1}] \leq q_{rt} + M(1 - y_{rt}), \forall r \in R_{s_i}, t \in T \tag{4}$$

$$\sum_{k=1}^{i} \ell_{r s_k} [x_{s_k t} + \sum_{j=2}^{u_{r s_k}} x_{s_k t}] \leq M y_{rt} + q_{rt} + v_{rt}, \forall r \in R_{s_i}, t \in T \tag{5}$$

$$x_{s_i t} \in \{0, 1\}, \qquad \forall t \in T \tag{6}$$

$$y_{rt} \in \{0, 1\}, \qquad \forall r \in R_{s_i}, t \in T \tag{7}$$

The objective function (1) consists of two terms, namely the penalty factor c_1 when time window I is not satisfied and the penalty factor c_2 when at least one resource r required by surgery s_i uses additional capacity. Equation (2) ensures that upon the existence of a feasible solution, the surgery is appointed to a day of the planning horizon T. Equation (3) ensures that a surgery is not scheduled before the start of the time window $\underline{I}$. Inequalities (4) and (5) represent the common notation for disjunctive constraints and describe whether the surgery's requirement for resources $r \in R_{s_i}$ can be satisfied under consideration of the previously reserved capacities for surgeries $s_1, \ldots, s_{i-1}$. In particular, Inequalities (4) are soft constraints since the capacity may be expanded by the parameter v_{rt} as in Inequalities (5), where the auxiliary variable y_{rt} indicates which of the two inequalities is binding for a particular r and t. Finally, Relations (6) and (7) define the domain of all decision variables as being binary.

The use of the above formulation presents a close relation to practice relevant criteria when determining an appointment for a surgery. Due to medical and patient preferences, an appointment is searched within the desired time window I. Such an appointment will be selected taking into account that a low over- and under-utilization of resources is desired, that hospitalization bed use is levelled, that no staff overtime is incurred

and that such a solution can be found in real time. Based on field work carried out in several German hospitals, satisfying the desired time window is more important than incurring permissible overtime (i.e. $c_1 > c_2 > 0$). The proposed solution method described in the next section makes use of this fact and of the structure of the feasible space of the TOTSP to find an optimal and practice relevant solution using a hybrid algorithm based on simple bin packing rules.

3 Solving the TOTSP

The feasible space of the TOTSP can be divided into equivalence classes according to their objective values. All feasible appointment days within the time window I that can be assigned to a surgery without any resources incurring overtime have the same objective function value, namely zero. Furthermore, all feasible appointment days within the time window and where at least one resource incurs overtime, have the same objective function value, namely c_2. Likewise, feasible appointment days lying outside I and that do not require overtime for any of the resources, have the same objective function value, namely c_1. Finally, feasible appointment days that lie outside I and which require overtime for at least one of the surgery's required resources, have an objective function value of $c_1 + c_2$. Based on this structure of the feasible space, finding an optimal solution for the TOTSP consists in selecting one solution amongst those solutions in the best equivalence class. Such a selection will be carried out according to the practical situation arising in each instance of the TOTSP.

Finding a solution within the first equivalence class will be done either with a FF or with a "best fit" (BF) strategy. The parameter that triggers either strategy is the desired time window I. Instances vary depending on its size and how immediate the start of the interval is. Since unused capacity corresponding to immediate days is lost when these become part of the past, it is important that days in the immediate future are filled to capacity. The FF strategy thus looks for the first appointment day that does not require overtime for any of the resources within candidate days belonging to the set $\mathcal{FF} = I \cap T_{FF}$, assuming that $\mathcal{FF}$ is non-empty and where T_{FF} is a hospital dependent and predefined number of days in the future that are desired to be filled. If such a solution is found, it will be optimal as it belongs to the first equivalence class. Otherwise, a BF strategy will be applied to the set of candidate appointment days $\mathcal{BF} = I \cap {T_{FF}}^C$. The BF strategy searches for each day in the set $\mathcal{BF}$, the resource $r \in R_s$ with

the tightest fit. For the day to be eligible, no overtime is incurred for any of the resources. The BF strategy then selects the appointment day as the day $t \in \mathcal{BF}$ that has the resource with tightest fit. If there are several candidate days with the same tightest fit, a tie breaking rule is applied according to the sum of squares (SSq) rule. The SSq rule analyzes the bed distribution for a portion of the days of the planning horizon T, say $\tilde{T} = [\underline{I} - p, \overline{I} + q]$, with p, q parameters selected in such a way that $\tilde{T} \subseteq T$. The optimal solution is then obtained by finding the solution to the following expression (and assuming that β_t represents the number of available beds on day t):

$$t^* = argmin_{t \in \tilde{T}} \Big\{ \sum_{t \in T} \beta_t^2 + \sum_{t \in \tilde{T} \setminus \{\overline{I}+q\}} (\beta_{t+1} - \beta_t)^2 \Big\} \tag{8}$$

If no feasible solution is found using either the FF or the BF strategy, then the first equivalence class is empty and solutions belonging to the second equivalence class (time window I is fulfilled and overtime is incurred for at least one of the required resources) will be optimal. Assuming that there exist feasible solutions, the optimal solution is thus found by selecting the day in time window I with the lowest incurred overtime as a result of the soft capacity constraints in the TOTSP. Should the equivalence class be empty, then the optimal solution will belong to the third equivalence class, which includes those solutions where the time window I is violated and no overtime is incurred. For this, a FF strategy is employed to find the first day after $\overline{I}$ for which an appointment can be found and all resources do not require the use of overtime. Only in the case that this third equivalence class is empty, will a FF strategy be required to find an appointment after $\overline{I}$.

4 Computational Experience

The proposed hybrid method was implemented in C++ and solved on a Pentium 4 PC with a 1.7 GHz processor and 512 MB RAM. Randomly generated instances were created based on a pool of 18 frequent surgeries in a mid size hospital in Germany. These instances consisted of patient arrival over a course of $11-36$ weeks and requiring a surgery appointment within a time horizon of six months. Four surgery rooms, 21 staff members (surgeons, anesthesiologists), one nursing ward with $21-35$ beds and one intensive care unit with nine beds were modelled. A surgery team consisted of at least two members and at most four. Finally, the set T_{FF} corresponded to the next week.

The results of the proposed hybrid approach are compared in Table 1 with the FF strategy commonly used in practice. It can be seen that the proposed solution approach performs better than the classical FF approach. Moreover, on average, an appointment was found within 0.3 seconds with the new method.

Table 1. Results of the hybrid and FF strategy to solve the TOTSP

Instance	Satisfaction of I (%)		Avg. deviation from I (days/surgery)		Overbooking of ORs (total hours)	
	Hybrid	FF	Hybrid	FF	Hybrid	FF
I-1	93	89	0,48	0,77	174	235
I-2	79	76	2,24	2,43	224	266
I-3	53	48	8,69	10,65	262	303
I-4	86	83	1,45	1,77	43	59
I-5	81	76	2,08	2,56	42	65
I-6	87	84	1,34	1,70	42	59
I-7	96	93	0,38	0,59	17	31
I-8	93	91	0,70	0,67	22	31
I-9	77	69	1,53	2,82	130	133
I-10	69	62	2,46	3,98	118	115
I-11	86	83	1,57	1,95	27	27

5 Conclusions and Outlook

The TOTSP formulation and solution method proposed in this paper support the process of finding an appointment for a surgery and yield a solution in real time. The model considers practice relevant aspects like a desired time interval during which the surgery has to be carried out, thus keeping waiting times within grasp. It utilizes resources efficiently which allows downstream levels of planning (next-day surgery scheduling and online scheduling) to return larger overall improvements in comparison to common practices in hospitals. Possible extensions to the model include allowing previously fixed appointments to be rescheduled to other days or to collect all incoming appointment requests during the course of a certain time period (e.g. one day) and then assign the appointments accordingly.

References

1. Macario A, Vitez TS, Dunn B, McDonald T (1995) Where are the cost in perioperative care? Aneshtesiology 83(6): 2–4

Part XIII

Managerial Accounting and Auditing

Modeling and Analyzing the IAS 19 System of Accounting for Unfunded Pensions

Matthias Amen

Katholische Universität Eichstätt-Ingolstadt, Auf der Schanz 49, DE-85049 Ingolstadt, Germany. Matthias.Amen@web.de

1 General Research Question

In the accounting system of the International Financial Reporting Standards (IFRS) pension plans are covered by the International Accounting Standard (IAS) 19 "Employee Benefits". As "accounting for defined contribution plans is straightforward" (IAS 19.43) we concentrate on defined benefit plans. In case of defined benefit plans, the employer has promised to make future pension payments according to a plan formula. We focus on pension plans without external accumulation of capital (unfunded plans) as we want to analyze pure accounting effects.

According to IAS 19.48 "accounting for defined benefit plans is complex". Due to experience deviations from the financial and non-financial assumptions actuarial gains and losses will occur. IAS 19.92-93A offers several alternative options to cope with actuarial gains and losses. IAS 19.95 assumes that actuarial gains and losses will offset in the long run. Because of the long horizon, the difficult calculations, the different additional elements in the system and the probabilistic nature, human expectations and heuristic approaches generally fail [3]. We are interested in the ability of offsetting actuarial gains and losses and investigate the effects of selected different options of coping with actuarial gains and losses.

2 A Brief Overview of the IAS 19 System

The defined benefit obligation (DBO) has to be measured by the *projected unit credit method* (IAS 19.64). For short, the DBO for a single individual is the present value of the expected future pension payments that are "earned" by the cumulated past work of the individual up to

the current balance sheet date according to a plan formula. For the calculation of the obligation we have to assume the following major parameters:

- expected future salary/wage increases
- expected future pension increases
- expected future fluctuation
- expected future mortality

IAS 19.78 requires to determine the discount rate from current market yields of long-term highly quality corporate bonds. The financial assumptions have to be mutually compatible (IAS 19.75).

The actuarial gains and losses derive from experience deviations from the actual to the expected values of these parameters. To some extend they are unavoidable, e.g. the realized mortality and fluctuation are either 0 or 1 and differ from the expected which are in the open interval (0, 1). IAS 19 provides alternative options to cope with actuarial gains and losses. We differentiate between four basic approaches:

1. cumulation and recognition of a corridor excess over the expected remaining working lives of the participating individuals (IAS 19.92)
2. cumulation and a faster recognition of a corridor excess, in particular recognition of a corridor excess in the current period (IAS 19.93, 93A)
3. immediate recognition in profit and loss (IAS 19.93, 93A).
4. immediate recognition outside profit and loss in a separate statement directly within equity (IAS 19.93A) (*equity approach*)

No. 1 and 2 are variations of the *corridor approach*. For the considered unfunded pension plans, the corridor is 10 % of last years DBO (IAS 19.92). In the following we concentrate on options No. 1 (which we call the "standard" corridor approach) and 4.

As the corridor approach allows to defer actuarial gains and losses, we have to differentiate between the DBO and the Defined Benefit Liability (DBL). Only the DBL will be shown in the balance sheet. Applying the corridor approach, the transition of the DBO, the cumulated unrecognized actuarial gains and losses, and the DBL from balance sheet date t-1 to t derives from the pension cost and the pension payments in the period t (Fig. 1). In the corridor approach the pension cost in period t are set up by the interest cost on the DBO_{t-1}, the current service cost in t (i.e. the present value of expected additional future pension payments due to the work in period t), and the actuarial gains and losses recognized in period t.

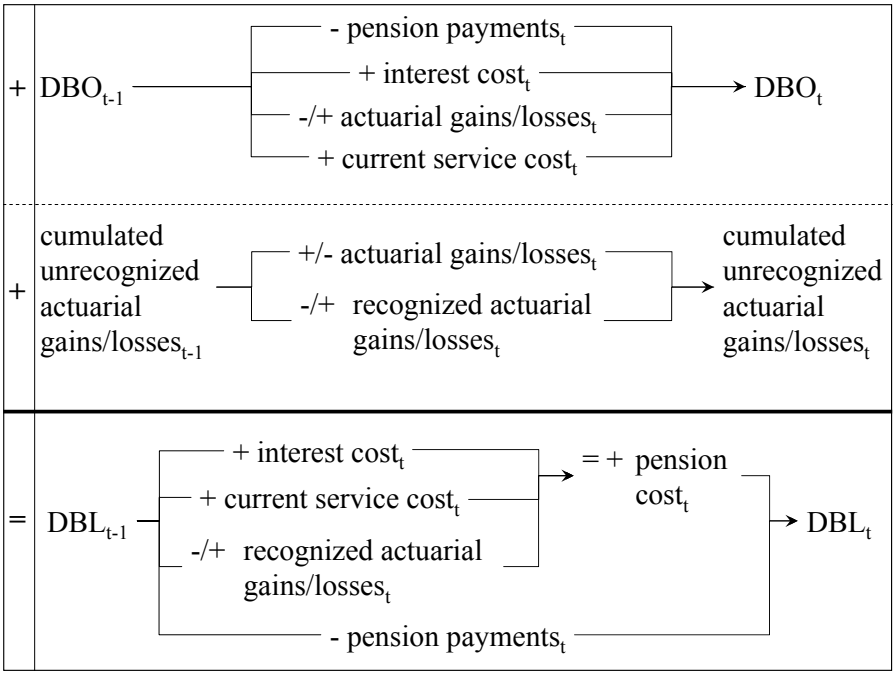


Fig. 1. Transition of elements in the corridor approach from t-1 to t

In the equity approach the DBL is identical to the DBO as there are no unrecognized actuarial gains and losses. The actuarial gains and losses are not part of the pension cost.

3 General Structure of the Simulation Model

Because of the complexity of the system an analytic solution to answer the question on offsetting actuarial gains and losses is impossible. Therefore, we use the Monte Carlo simulation technique in a discrete simulation model. In this paper we focus on the *degenerating* workforce version of the model, i.e. the initial workforce declines because of retirement, fluctuation or death of the employees. In the discrete model we work with a time interval of one year for a projection horizon of 88 years. The model has been implemented in Microsoft Excel by use of the Add-In Crystal Ball. We assume that the first pension payment for a retiree is calculated as a fixed percentage for each active year of the last salary before retirement (a final pay plan).

The simulation model consists of a valuation model at state t and a stochastic transition from state t to state t+1 (Fig. 2). The mortality

and the fluctuation have been implemented as binomially distributed events.

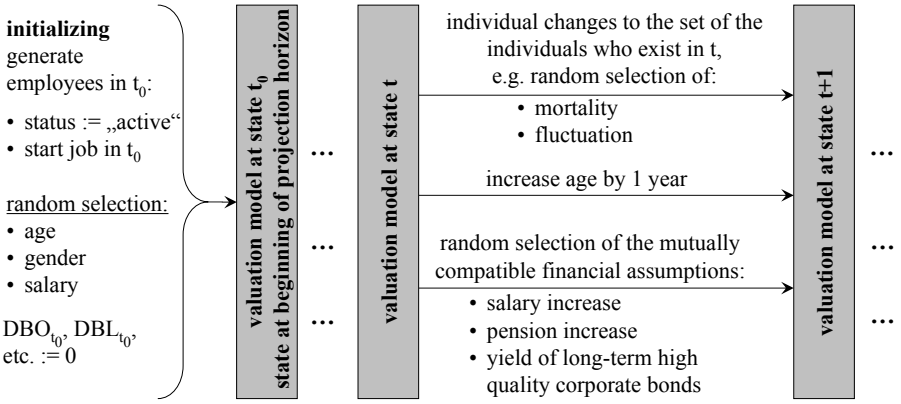


Fig. 2. General structure of the simulation model

The assumptions concerning the parameters are based on official German statistics (for details see [1]). We assume that the annual pension increase follows the non-negative inflation rate. Because of missing data, the yield of long-term high quality corporate government bonds has been modeled by a combination of related time series. Especially for the long-term projection of financial parameters we explicitly consider the correlations and autocorrelations. Based on the annual data of the financial parameters for 1957-2003, we generated a vector-autoregressive model of order 1 – VAR(1) – (see [2]) and integrated it into the simulation model.

$$\mathbf{y_t} = \mathbf{B} \times \mathbf{y_{t-1}} + \boldsymbol{\beta_0} + \mathbf{u_t}$$

with

$$\mathbf{y_t} = \begin{pmatrix} \text{inflation}_t \\ \text{salary increase}_t \\ \text{yield}_t \end{pmatrix} \mathbf{B} = \begin{pmatrix} 0.433 & 0.162 & 0.396 \\ -0.644 & 0.904 & 0.515 \\ 0.016 & 0.089 & 0.677 \end{pmatrix} \boldsymbol{\beta_0} = \begin{pmatrix} -2.128 \\ -1.382 \\ 1.730 \end{pmatrix}$$

The residuals $\mathbf{u_t}$ are normally distributed with the mean 0 and the standard deviations 0.86 (inflation), 1.90 (salary increase), and 0.96 (yield). The Bravais/Pearson correlation coefficients between the residuals are 0.18 (inflation & salary increase), 0.50 (inflation & yield), and 0.24 (salary increase & yield). Despite the fact that the generation

of random numbers works with Spearman rank correlations these values have been used as parameters. There are only marginal differences between the assumed and the simulated Bravais/Pearson correlation coefficients. After an in-swinging phase of 20 years, finally we get a stationary and homoscedastic model with long-term expected values of 2.62 % (inflation), 5.30 % (salary increase), and 6.95 % (yield).

4 Fundamental Results

Financial reporting should provide useful information for investing decisions (F 12-21). The options to cope with actuarial gains and losses may cause biased information as some depend on the assumption of offsetting of actuarial gains and losses in the long run. This fundamental assumption can be examined best for the equity approach which cumulates the actuarial gains and losses separately within equity. In case of a degenerating workforce we get a final state at the end of the projection horizon.

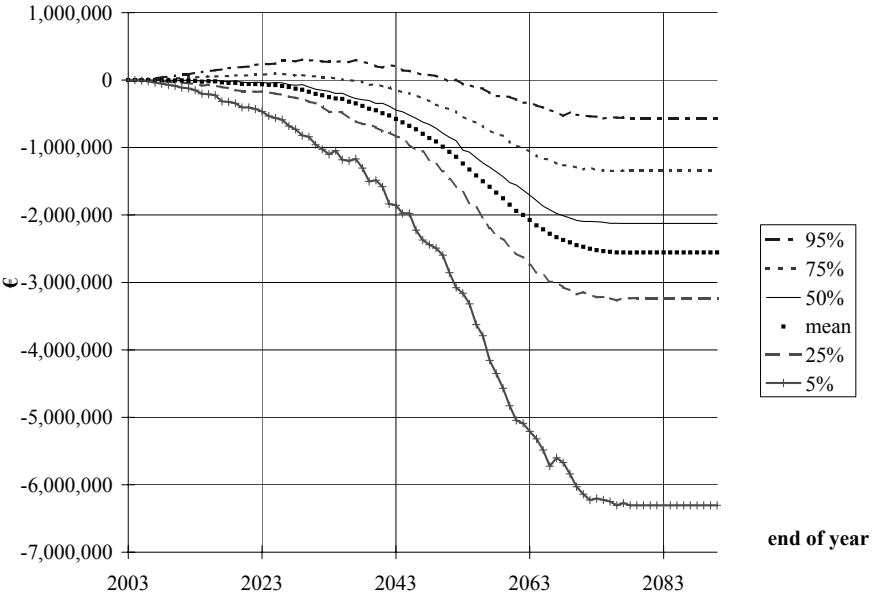


Fig. 3. Cumulated actuarial gains and losses - recognized directly in equity

As we apply the VAR(1)-model for projecting the financial parameters and make some minor modifications the numbers differ from those

in [1], but the general shape and outcome is identical (see Fig. 3). Therefore, the fundamental results can be taken as robust. It is obvious that there is only a poor chance of offsetting cumulated actuarial gains and losses in the long run. In case of the degenerating workforce on average 29.8 % of the total pension payments will never be recognized as pension cost if we decide to work with the equity approach.

If we apply the standard corridor approach, then the line of the mean for the cumulated unrecognized actuarial gains and losses is shaped like a bowl. After the initial downward trend of the mean, it improves from the mid of the projection horizon. Finally the mean is 0. Compared with the equity approach the downward trend is buffered by the fact that there is an amortization of the corridor excess. The subsequent upward trend can be explained mathematically. Because of the decline of population and remaining lifetime, the DBO and the corridor decreases. An (larger) excess of the (smaller) corridor will be amortized faster.

5 Conclusions

The major result is the fact that cumulated actuarial gains and losses are not symmetrical. Especially the equity approach causes a systematic and permanent bias in accounting information. As the International Accounting Standards Board (IASB) has announced to develop a new standard by 2010, we strongly recommend rejecting the equity approach. Finally we prefer an immediate recognition of any actuarial gains and losses – or, at least, the corridor approach that avoids a permanent bias.

References

1. Amen M (2007) Simulation based comparison of existent IAS 19 accounting options. European Accounting Review 16:243–276
2. Lütkepohl H (2005) New Introduction to Multiple Time Series Analysis. Springer, Berlin Heidelberg New York
3. Tversky A, Kahneman D (1974) Judgment under uncertainty: heuristics and biases. Science 185:1124–1131

Coordination of Decentralized Departments and the Implementation of a Firm-wide Differentiation Strategy

Christian Lohmann

Institute of Production Management and Managerial Accounting, Ludwig-Maximilians-University Munich, 80539 Germany. lohmann@lmu.de

1 Introduction

This study regards a two-stage value chain of a decentralized company. The separated investment centers can implement a firm-wide differentiation strategy by making specific investments. The focus is set on a situation where the upstream manufacturing department invests in product quality improvements and the downstream marketing department invests in marketing operations. The specific investments are totally defrayed by the units acting on their own authority. Because both specific investments affect the whole revenue in the same way increasing customers demand and customers willingness to pay, the allocation of the profit induced by the specific investments is not made cost reflective. An underinvestment problem arises, witch endangers the objective of firm-wide profit maximization.

This article compares a contribution margin sharing rule, a revenue sharing rule and a transfer pricing scheme as coordination instruments for achieving goal congruence between the departments inducing efficient production and investment decisions as well as overall firm profit maximization. Contribution margin or profit sharing (e.g. [3]) and transfer pricing (for an overview see [4]) are well known. In contrast, revenue sharing is an as far as possible unknown instrument for coordinating decentralized companies. Starting points for considering revenue sharing systems and the effects of revenue affecting specific investments in a decentralized company are shown by Chwolka/Simons [1] and Martini [2].

The remainder of this paper is organized as follows: Section 2 presents the framework and the solution of an equilibrium model using a contribution margin sharing rule, a revenue sharing rule and a

transfer pricing scheme. In section 3 the performance of these coordination instruments is compared on the basis of the overall firm profit. Circumstances are identified, under witch a single coordination instrument dominates the others.

2 Model

2.1 Assumptions

The considered company consists of two decentralized departments organized as investment centers. At the first stage investment center A produces an intermediate product at constant unit cost c_A. After transfer to investment center B the intermediate product is completed at constant unit cost c_B and sold on an anonymous market. B is in charge for the sales volume x, A adapts the sales volume generating a nonnegative profit. The risk-neutral investment center managers are compensated according to their reported success after profit allocation by using a contribution margin sharing rule, a revenue sharing rule or a transfer pricing scheme.

The assumed multiplicative demand function with constant price elasticity $\epsilon = 2$ is given by

$$p(x, I_A, I_B, \tilde{a}) = \sqrt{\frac{\tilde{a}(v\sqrt{I_A} + w\sqrt{I_B})}{x}} \qquad \text{with the attributes}$$

$$\frac{\partial p(x, I_A, I_B\, \tilde{a})}{\partial x} < 0 \quad , \quad \frac{\partial p(x, I_A, I_B, \tilde{a})}{\partial I_i} > 0 \quad \text{and}$$

$$\frac{\partial^2 p(x, I_A, I_B, \tilde{a})}{\partial^2 I_i} < 0 \quad , \quad \text{for} \quad i \in A, B \quad \text{as well as} \quad \frac{\partial^2 p(x, I_A, I_B, \tilde{a})}{\partial I_A\, \partial I_B} = 0\,.$$

The demand function depends on the sales volume, the specific investments I_A and I_B as well as the random variable $\tilde{a}$. Both investment centers can support a firm-wide differentiation strategy by making specific investments. The specific investments affect the demand function with different efficiencies v and w. With $v > w$, the influence of I_A on the demand function is larger than I_B. The random variable $\tilde{a}$ reflects uncertainty of market conditions at the selling moment. The random variable $\tilde{a}$ with mean $\mu = a$ implies that the expected price is given by

$$E[p(x, I_A, I_B, \tilde{a})] = \sqrt{\frac{a(v\sqrt{I_A} + w\sqrt{I_B})}{x}} \quad .$$

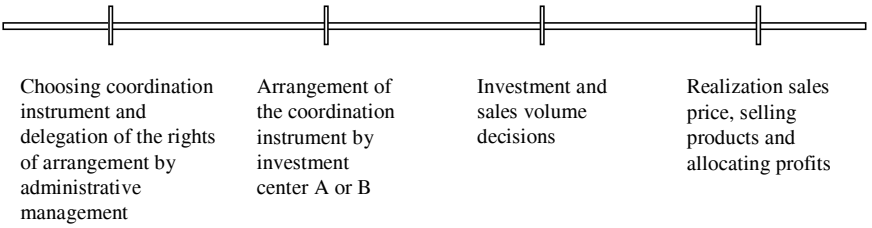


Fig. 1. Decision making and time sequence of the production process

The decision making and time sequence of the production process is pictured in fig. 1. At the outset, the administrative management of the company chooses a coordination instrument delegates the right of arrangement either to A or to B. Before investments and production take place, A or B determine the transfer payment by fixing the contribution margin share, the revenue share or the transfer price (monopolistic decision). Then, the investment centers make their specific investments. Without additional information B determines the sales volume and orders from A the corresponding quantity of the intermediate product. Finally, the market price is realized and the profits are allocated.

The specific investments are not observable and can not ascertain ex post because of random variable $\tilde{a}$. But each investment center manager knows the investment decision problem of the other manager. Therefore, they are able to calculate the optimal investment level in their view. The investment center managers are planning their decisions on the basis of their individually expected department profit. The following analysis uses an equilibrium model solving the several decision problems by reverse induction.

2.2 Solution

The points of departure are the expected profit functions of A and B using contribution margin sharing

$$\pi_A(x, I_A, I_B, \tau^{CM}) = \tau^{CM} \left(\sqrt{(v\sqrt{I_A} + w\sqrt{I_B})\, a\, x} - (c_A + c_B)x \right) - I_A$$

$$\pi_B(x, I_A, I_B, \tau^{CM}) = (1-\tau^{CM}) \left(\sqrt{(v\sqrt{I_A} + w\sqrt{I_B})\, a\, x} - (c_A + c_B)x \right) - I_B,$$

revenue sharing

$$\pi_A(x, I_A, I_B, \tau^{RS}) = \tau^{RS} \sqrt{(v\sqrt{I_A} + w\sqrt{I_B})\, a\, x} - c_A x - I_A$$

$$\pi_B(x, I_A, I_B, \tau^{RS}) = (1 - \tau^{RS}) \sqrt{(v\sqrt{I_A} + w\sqrt{I_B})\, a\, x} - c_B x - I_B$$

or transfer pricing

$$\pi_A(x, I_A, T^{TP}) = T^{TP}\, x - c_A\, x - I_A$$
$$\pi_B(x, I_A, I_B, T^{TP}) = \sqrt{(v\sqrt{I_A} + w\sqrt{I_B})\, a\, x} - (c_B + T^{TP})x - I_B.$$

B specifies the optimal sales volume depending on the investment levels I_A and I_B as well as the allocation parameter. Anticipating the quantity decision the investment centers determine their specific investments depending on the allocation parameter. Finally, either A or B set the allocation parameter maximizing their department profit regarding contribution margin sharing

$${}^{A}\tau^{CM} \in \left\{\frac{w^2}{2w^2 - v^2}, 1\right\} \qquad {}^{B}\tau^{CM} \in \left\{0, \frac{v^2 - w^2}{2w^2 - v^2}\right\},$$

revenue sharing

$${}^{A}\tau^{RS} = \frac{2v^2c_A^2 + 5v^2c_Ac_B - 4w^2c_Ac_B + 2v^2c_B^2 - 5w^2c_B^2}{2\,(v^2c_A^2 + 4v^2c_Ac_B - 2w^2c_Ac_B + 4v^2c_B^2 - 4w^2c_B^2)} + \frac{c^B\sqrt{v^4c_A^2 + 4v^4c_Ac_B - 2v^2w^2c_Ac_B + 4v^4c_B^2 + 9w^4c_B^2}}{2\,(v^2c_A^2 + 4v^2c_Ac_B - 2w^2c_Ac_B + 4v^2c_B^2 - 4w^2c_B^2)}$$
$${}^{B}\tau^{RS} \in \left\{\frac{2v^2c_A + v^2c_B - w^2c_B}{2v^2c_A + 4v^2c_B - w^2c_B}, \frac{c_A}{c_a + 2c_B}\right\}$$

or transfer pricing

$${}^{A}T^{TP} = \frac{3c_A(v^2 + w^2) + (v^2 - w^2)c_B + (c_A + c_B)\sqrt{v^4 + 2v^2w^2 + 9w^4}}{2\,(v^2 + 2w^2)}$$
$${}^{B}T^{TP} \in \left\{c_A, \frac{3v^2c_A + (v^2 - w^2)c_B}{2v^2 + w^2}\right\}.$$

Using forward induction the decision interdependencies can be resolved and the investment levels, the sales volume, the expected department profits as well as the expected overall firm profit can be calculated.

3 Performance Evaluation

The expected overall firm profit reflects the ability of the coordination instruments inducing efficient investment and sales volume decisions. Therefore, the coordination instruments are compared on the basis of the overall firm profit. Fig. 2 shows the overall firm profit depending on relative unit costs and relative efficiencies of the specific investments

for contribution margin sharing, revenue sharing and transfer pricing determined either by A or B.

Depending on relative unit costs and relative efficiencies of the specific investments the implementation of every single coordination instrument determined either by A or B can be expedient. The relation of the efficiencies of the specific investments has a greater impact on the performance of the coordination instruments than the relation of the unit costs. Conspicuous are three areas dominating by a single coordination instrument.

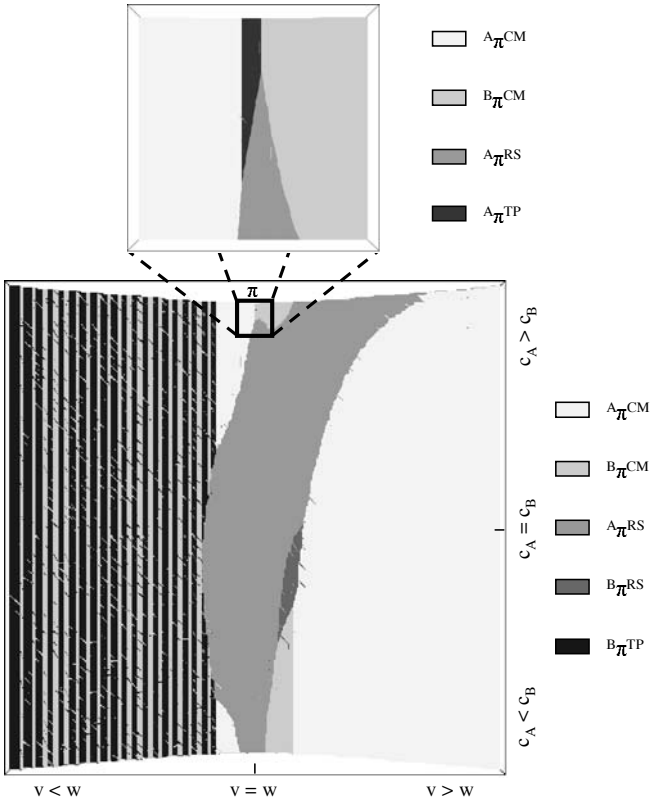


Fig. 2. Overall firm profit depending on relative unit costs and relative efficiencies of specific investments with $a = 10$, $c_A \in]0,2[$, $c_B \in]0,2[$, $c_A + c_B = 2$, $v \in]0,2[$, $w \in]0,2[$ and $v + w = 2$ (top view with blow up)

1. $v < w$: The influence of I_A on the demand function is larger than I_B. Contribution margin sharing or transfer pricing by B cause the

highest expected overall firm profit. In that case, B sets ${}^{B}\tau^{CM} = 0$ or ${}^{B}T^{TP} = c_A$. A makes no specific investments ${}^{B}I_A^{CM} = {}^{B}I_A^{TP} = 0$ and realizes the expected department profit ${}^{B}\pi_A^{CM} = {}^{B}\pi_A^{TP} = 0$. Only B makes specific investments. Because of the weak influence of I_A the investment decisions are reduced to a single investment decision of B. The expected overall firm profit corresponds with the expected department profit of B.

2. $v > w$: The influence of I_A on the demand function is lower than I_B. The highest expected overall firm profit is generated by A setting the contribution margin share at ${}^{A}\tau^{CM} = 1$. The hole contribution margin is allocated to B. Concerning contribution margin sharing the sales volume decision of B is independent of its contribution margin share. In this particular case, only B makes specific investments and shows the hole expected profit.
3. $v \approx w$: Revenue sharing by A induces the highest expected overall firm profit, if the influence of I_A on the demand function is comparable with I_B. A and B make specific investments and implement as a result a firm-wide differentiation strategy. Both expect a department profit.

In the cases $v < w$ and $v > w$, efficiencies of the specific investments are so different that only the specific investment with the higher efficiency is implemented. If both specific investments have an important effect on the expected overall firm profit, revenue sharing performs best. In contrast to that, transfer pricing is not qualified for improving the investment tendency of the upstream investment center. Investment center A can only profit from a larger sales volume but not from a higher sales price. Hence, transfer pricing should not be used for implementing a firm-wide differentiation strategy.

References

1. Chwolka A, Simons D (2003) Impacts of Revenue Sharing, Profit Sharing and Transfer Pricing on Quality-Improving Investments. European Accounting Review 12:47-76
2. Martini JT (2007) Verrechnungspreis zur Koordination und Erfolgsermittlung. Deutscher Universitäts-Verlag, Wiesbaden
3. Milgrom P, Roberts J (1992) Economics, Organization and Management. Prentice Hall International, New Jersey
4. Pfaff D, Pfeiffer T (2004) Verrechnungspreise und ihre formaltheoretische Analyse. Zum State of the Art. Die Betriebswirtschaft 64:296-319

Case-Based Decision Theory: An Experimental Report

Wolfgang Ossadnik and Dirk Wilmsmann

University of Osnabrück, School of Economics and Management,
Department of Managerial Accounting
wolfgang.ossadnik@uos.de, dirk.wilmsmann@uos.de

1 Introduction

Theoretical models as Expected Utility Theory (EUT) start from the premise that decision making individuals follow subjective rules which they found e. g. on their personal probability judgements. Thus, EUT requires that the decision maker has a complete knowledge of all relevant states of the world and consequences of possible actions. Often, there is a lack of information basis to attribute probabilities to states of the world and to evaluate the utility of potentially occurring consequences. In such cases, alternative decision theories to EUT are to be applied for a reality adequate explanation of decision-making processes. Such a theory, being presented in the following, is Case-Based Decision Theory (CBDT), which will be analysed experimentally with respect to its empirical validity. The results shall substantiate the validity of the premises and the methods of the CBDT in the context of a repeated-choice problem.

2 Funding Repetitive Decisions by Case-Based Decision Theory

In a multitude of decision problems, individuals observably do not construct probabilities for potential states of the world, but tend to evaluate their actual decision problem with information about past decision situations. Based on this paradigm of human behaviour, CBDT postulates that individuals tend in their decisions under uncertainty to action alternatives having led to desirable consequences in the past. In this, the decision maker of CBDT refers to decision situations he/she

has been confronted with in the past, instead of referring to states of the world (cf. [3] and [5] in the following).

Reconstructing the subjective system of objectives in a case, CBDT provides the starting premises for the decision making process. Formally, such a case can be represented by a problem $q \in \mathcal{Q}$, a possible action $a \in \mathcal{A}$ and a resulting consequence $r \in \mathcal{R}$. The decision criterion is composed of a function $s : \mathcal{Q} \times \mathcal{Q} \rightarrow [0,1]$ and a utility function $u : \mathcal{R} \rightarrow \mathrm{IR}^+$. In general the function $s(\cdot)$ is called similarity function, because it expresses the similarity perceived by the decision maker between two problems. In the case of an identical perception of both problems, the similarity function $s(\cdot)$ takes the value one, while it takes the value zero in case of entirely different evaluated problems.

Additionally, an aspiration level is considered as differentiation of the decision criterion in order to display ambitious and at the same time careful decision behaviour. To determine the aspiration level, various adjustment rules are suggested (cf. [4]). In the following, let the function $c_t(a)$ describe the number of previous periods in which the act a was chosen. The average utility $d_t(a)$, accruing for the decision maker when the act a is chosen, can be formally determined using the relation

$$d_t(a) = \frac{\sum_{\tau \in c_t(a)} u(r_\tau)}{c_t(a)}, \text{ where } c_t(a) > 0 \tag{1}$$

In that, it is postulated that when considering an aspiration level, the utility of the action alternative a chosen should not be smaller than the utility of the action alternative(s) $\tilde{a}$ not chosen. For more than two alternatives to be evaluated, the result is consequently a linear system of equations whose solution is described by a set of feasible aspiration levels in the form of $\alpha_t \in [\underline{\alpha}_t, \overline{\alpha}_t]$. In the case of there being no solution to the system of equations, the aspiration level should take on the value of the previous period. The assumptions can be summarised formally using the following heuristics:

$$\alpha_t = \begin{cases} \alpha_0 & \text{if } t = 1 \\ \max\limits_{a \in \mathcal{A}} d_t(a) & \text{if } t \geq 2, c_t(a) > 0 \text{ and } \alpha_t \in [\underline{\alpha}_t; \overline{\alpha}_t] \\ \underline{\alpha}_t & \text{if } t \geq 2, c_t(a) > 0 \text{ and } \alpha_t < \underline{\alpha}_t \\ \overline{\alpha}_t & \text{if } t \geq 2, c_t(a) > 0 \text{ and } \alpha_t > \overline{\alpha}_t \\ \alpha_{t-1} & \text{else} \end{cases} \tag{2}$$

Are these requirements cumulatively met, the utility of an action can be determined as follows:

$$U_{t,\alpha}(a) = \sum_{\tau=1}^{t-1} s(q_t, q_\tau) \cdot [u(r_\tau) - \alpha_t] \tag{3}$$

In this situation of uncertainty it is to be analysed, whether different models of behaviour are in a comparable or stronger way empirically valid than the behavioural hypotheses of CBDT. Therefore it shall be investigated in the following, if EUT can explain decision behaviour experimentally observed. Often EUT is considered as an approximation of economical behaviour (cf. among others [2], 66 ff.).

Application of the EUT requires the existence of a risk situation, i. e. of probabilities of action consequences. As the Bayesian actor does not know the specific distribution of the consequences of the various action alternatives in this situation of uncertainty, he regards the probabilities of consequences $p(r)$ as randomised and therefore models them on the basis of their frequency $m(r)$. Is no information available and are no action alternatives realised, the Bayesian actor assumes a rectangular distribution of the consequences. Thus, the situation of uncertainty is interpreted as a situation of risk. The idea of a distribution of consequences being considered in textbooks as adequate, has to be successively adjusted with realisation of the consequences (cf. [1]). On the basis of a sequence of choices, the development of the probabilities of the consequences can be determined with (4) in which n is the number of possible consequ! ences:

$$p_a(r_i) = \frac{m(r_i) + 1}{\sum_{i=1}^{n} m(r_i) + n} \tag{4}$$

The actor will evaluate the available action alternatives on the basis of (5) by weighting the possible consequences with probability resulting from (4):

$$E[U(a)] = \sum_{i=1}^{n} p_a(r_i) \cdot u(r_i) \tag{5}$$

On the basis of the observed decision behaviour of students of Management and Economics at the University of Osnabrück (Germany) aspects of the formal structure of the evaluation calculus of CBDT and of its empirical validity in comparison to the validity of EUT shall be discussed in the following.

3 An Experimental Study on Case-Based Decision Theory

On account of its simple variability and its intelligible structure, an urn experiment is conducted. It is divided into three passes that differ with regard to the number and composition (resp. distribution) of the balls to be drawn. Each ball shows three colour fields ("red", "black" and "blue") within which whole numerical values as an element of the set $[-5, 5]$ are written, so that altogether for each ball there are a total of 11^3 possible colour-number combinations. The test persons know merely the number of balls contained in the urn. However, they have no knowledge about the number-colour combinations of the individual balls. The task of the test persons is to choose a colour a_i. After selecting a colour, a ball is drawn randomly and the corresponding numerical value is announced, that is $r(a_i) = r_i \in [-5, 5]$. The test person is not given any further information about the composition of the ball, that is, the two remaining co! lour-number combinations. The ball is returned to the urn immediately after the drawing. In this respect, each of the three passes describes a so-called repeated-choice situation.

In the beginning situation (Pass A) the urn is stocked with nine balls. After 15 drawings, three balls are removed from the urn without announcing their composition to the experiment participants. Again, 15 drawings (Pass B) take place. After the 30th drawing, five balls whose composition is not known by the participants are added to the six balls already in the urn. Another 15 drawings are carried out (Pass C).

A total of 269 test persons takes part in the experimental study. In order to avoid arrangements being made among the test persons with regard to a common game strategy, the test persons are divided into five groups by a random generator, and each of these groups stands for a specific stocking of the urn (or writing on the ball). In reconstructing the decision behaviour of the test persons, there are, in addition, the following assumptions: the outcome values correspond at the same time to the partial utility values $u(r_i) = r_i$ and the similarity measure $s(\cdot)$ is constant and equal to one within one pass. This means, that the test person notices the identity of the decision problem, that is, the repeated-choice situation as such.

Providing the decision principles of CBDT (3) and of EUT (5), the percentages of conformity between observed decision behaviour and theoretical behaviour hypotheses result as shown in Tab. 1, which differentiates in five groups, in single decisions, and in decision sequences.

Table 1. Percentages of observed decisions conforming to theoretical behaviour hypotheses

	Group 1	Group 2	Group 3	Group 4	Group 5	All
Single Decisions						
CBDT $[U_{t,\alpha}(\cdot)]$	82.65	89.18	88.09	92.33	88.99	88.73
EUT $[E[U(\cdot)]]$	49.33	61.22	53.07	58.80	64.23	57.51
Decision Sequences						
CBDT $[U_{t,\alpha}(\cdot)]$	23.64	16.98	7.84	25.00	9.68	16.36
EUT $[E[U(\cdot)]]$	0.00	3.74	0.00	4.17	1.61	1.86

The percentages of empirically observed behaviour (single decisions) conforming to the theoretical hypotheses evidently deviate between CBDT (88.73%) and EUT (57.51%). By considering sequences of decisions, only 1.86% of the observed decision behaviour conforms to EUT, whereas 16.36% conforms to CBDT.

We sum up that in the context of our experiment CBDT has a higher empirical validity than EUT. Thus our results confirm the hypothesis, that decision making on the basis of similarity measures $s(\cdot)$ and under consideration of an aspiration level α describes a decision process being intuitive and plausible but also empirically valid.

4 Conclusion and Outlook

The results of our experimental study prove that CBDT can contribute to the explanation of subjective decision behaviour. However, the reliability of results is linked to the design of the study and the validity of the (partially restrictive) assumptions regarding the heuristics used as a basis. Future research must therefore gradually extend the basic conditions of our experimental study, and analyse the functional structure of similarity measures and the development of aspiration levels in a strengthened manner. Provided the decision premises of CBDT can be confirmed in further empirical studies, its prescriptive application becomes possible and CBDT can be considered in economic model construction and analysis. Altogether, it can be attested that CBDT opens possibilities of looking into economic problems in a new, more strongly intuitive way, integrating cognitive restrictions of decision makers in economic models.

References

1. Clemen RT (1996) Making Hard Decisions. Duxbury Press, Pacific Grove et al.
2. Fischer K (2004) Aspekte einer empirisch fundierten betriebswirtschaftlichen Entscheidungslehre, Gabler, Wiesbaden
3. Gilboa I, Schmeidler D (1995) Case-Based Decision Theory. The Quarterly Journal of Economics 110(4):605–639
4. Gilboa I, Schmeidler D (1996) Case-Based Optimization. Games and Economic Behavior 15(1):1–26
5. Gilboa I, Schmeidler D (2001) Case-Based Decision Theory. Cambrigde University Press, Cambridge
6. Guerdjikova A (2004) Case-Based Decision Theory and Financial Markets. MA Thesis, Universität Heidelberg

Part XIV

Multi Criteria Decision Making

Truck Allocation Planning for Cost Reduction of Mechanical Sugarcane Harvesting in Thailand: An Application of Multi-objective Optimization

Kriengkri Kaewtrakulpong[1], Tomohiro Takigawa[2], and Masayuki Koike[2]

[1] Faculty of Agriculture, Kasetsart University, Thailand. agrkkk@ku.ac.th
[2] Graduate School of Life and Environmental Sciences, University of Tsukuba, Japan

1 Introduction

In Thailand, the chopper-type mechanical sugarcane harvester is widespread. This type of harvester usually operates with some trucks. When sufficient numbers of trucks are available, the chopper can carry out continuous harvest operation. However, there are not enough trucks in the surveyed region to cover the transportation needs. Hence, effective allocation planning of trucks is vital. By determining the optimum number of trucks needed for the sugarcane fields, it is possible to make efficient use of the chopper.

Our field study found that there are three groups involved in these processes: sugarcane farmers, the owners of mechanized resources, and sugar factories. The problem of harvesting and transportation is crucial for all three. As well, they each have their own needs in the harvesting and transportation processes. Both the owners of mechanized resources and sugarcane farmers would like to minimize the number of operating days required to harvest the fields, and the owners of trucks would also like to minimize the total traveling distance of trucks to reduce fuel costs. The sugarcane farmers and sugar factory want to minimize deterioration time of the harvests. Therefore, in this study, both efficiency and the appropriate distribution of profit were considered through multi-objectives planning reflecting the desires of all groups.

2 Data Sources and Simulation

The field investigation was performed from August through December, 2005 in Udon Thani province, northeastern Thailand. Field size, field shape, distance to a sugar factory, and geographical location of each sugarcane plot were measured. Time studies of the operation of mechanical harvesters and trucks were carried out.

Models simulating mechanical sugarcane harvesting and transportation in Thailand were developed based on [1]. For this simulation, the results obtained from our time studies and survey of trucks were used to determine the parameters for calculation of the transported amount of sugarcane by use of a truck, or a trailer from a certain field.

The truck loading time (TLT) is the time in minutes required to fill one truck. It can be given by:

$$TLT = \left(\frac{COT}{AOS}\right)\left(\frac{FS \times 10000}{RL \times RS}\right)\left(\frac{RL}{CSP} + TTT\right) + \frac{TTC}{60} \qquad (1)$$

, where COT is the capacity of the truck, ton; RL is the row length of the field, m; RS is the row spacing of the crop which is set equal to 1.5m; AOS is the amount of sugarcane, ton; FS is the field size, ha; CSP is the average cutting speed of the mechanical harvester, m/min; TTT is the turning time of the harvester and truck at the head land, min., and TTC is the time required for truck changing, seconds.

A round trip time involves the travel time from the field to the factory and the return time, as well as the amount of time the truck waits in a queue at the factory, and the time for reception and unloading of sugarcane at the factory, that is given by:

$$TRTR = \left(\frac{DMF}{ASF} + \frac{DMF}{ASE}\right) \times 60 + DPT + BDT + (WTM + TRO) \qquad (2)$$

Or the TRTR can be rewritten with the summation of TGO, TFACT, and TBACK; these terms are defined as follows.

$$TGO = \left(\frac{DMF}{ASF}\right) \times 60 + \left(\frac{DPT + BDT}{2}\right) \qquad (3)$$

$$TFACT = WTM + TRO \qquad (4)$$

$$TBACK = \left(\frac{DMF}{ASE}\right) \times 60 + \left(\frac{DPT + BDT}{2}\right) \qquad (5)$$

where TGO is the time consumed delivering the sugarcane from the field to the sugar factory, min; TFACT is the time consumed by the

truck at the sugar factory, min; TBACK is the return trip time of the truck from the sugar factory to the field, min; DMF is the distance from the field to the sugar factory, km; ASF is the average speed of a loaded truck, km/h; ASE is the average speed of an empty truck, km/h; DPT is the driver's personal time per truck per round trip, min; BDT is the time spent on refueling of the truck per round trip, min; WTM is the waiting time of the truck in queue at a sugar factory, min, and TRO is the time for reception and unloading operations at a sugar factory in minutes.

Prior to transport, the time for adjustment of the harvested sugarcane in the truck, TAD, was added approximately 5 and 10 minutes for a 10-wheeled truck and a trailer, respectively.

The deterioration time (DT) of the harvested sugarcanes could be estimated by using following equation:

$$DT = TAD + TGO + (TFACT - 5) \tag{6}$$

In this formula, the time for weighing the empty truck, around 5 minutes, was subtracted from the time the truck spends at the sugar factory.

The obtained values and the number of working-hours per day are required in order to determine the number of round trips per day that a 10-wheeled truck (STRIP) makes and that a 10-wheeled truck with a trailer (LTRIP) makes. Also, the amount of delivered sugarcane per day that a 10-wheeled truck (SD) makes, and that a 10-wheeled truck with a trailer (LD) makes can be determined. These values were used as the input parameters of the allocation plans to determine the optimum number of trucks for each sugarcane field.

3 Application of MOO to Allocate Mechanized Resources

3.1 Development of Allocation Plans

The number of 10-wheeled trucks and 10-wheeled trucks with trailers allocated in fields i, ST_i and LT_i respectively, are defined as the decision variables of the allocation plans.

Objective function 1: Minimize number of operating days
If the harvesting and transportation processes can be accomplished in a shorter period of time, the owners of mechanized resources can begin operations on fields waiting for harvest. Post-harvest operations, such as land preparation for the next crop, can also be begun earlier.

When assuming that one mechanical harvester is allocated to one field, minimization of the number of harvesting and transportation days can be expressed as

$$Minimize \sum_{i=1}^{h} \frac{Y_i}{(ST_i \times SD_i + LT_i \times LD_i)} \tag{7}$$

where h is the number of mechanical harvesters; Y_i is the yield of the field i in which the harvester i operates, ton.

The objective function is usually constrained by the availability of mechanized resources. It is also subjected to the daily milling capacity of the factory. Thus, the set of constraints can be expressed as follows.

$\sum_{i=1}^{h} ST_i \leq$ *Numbers of 10-wheeled trucks available for transporting the harvests*

$\sum_{i=1}^{h} LT_i \leq$ *Numbers of 10-wheeled trucks with trailers available for transporting the harvests*

$\sum_{i=1}^{h} (SD_i + LD_i) \leq$ *Maximum of daily amount of mechanical harvested sugarcanes supplied to the sugar factory*

$\sum_{i=1}^{h} (SD_i + LD_i) \geq$ *Minimum of daily amount of mechanical harvested sugarcanes supplied to the sugar factory*

Objective function 2: Minimize total traveling distance of trucks

Minimizing the distance is crucial due to the current high price of fuel. Thus, truck owners would like to minimize the total traveling distance of their trucks.

$$Minimize \sum_{i=1}^{h} (ST_i \times STRIP_i + LT_i \times LTRIP_i) DAY_i \times DIS_i \times 2 \tag{8}$$

where DAY_i is the amount of operating days of harvester i operating on field i, and DIS_i is the distance from the field i to the factory, km.

Objective function 3: Minimize deterioration time of the harvested sugarcanes

If this time is long, the loss in the weight and quality of the harvested sugarcane will increase [2]. This will result in a decrease in the income

of sugarcane farmers. This also leads to a reduction in the amount of sugar produced and thus in the income of the sugar factory.

$$Min. \sum_{i=1}^{h} (ST_i \times SDT_i \times STRIP_i \times DAY_i) + (LT_i \times LDT_i \times LTRIP_i \times DAY_i) \tag{9}$$

where SDT_i is the deterioration time of the sugarcane transported by a 10-wheeled truck worked with harvester i, min, and LDT_i is the deterioration time of sugarcane transported by a trailer worked with a harvester i, min.

3.2 Solution Method of MOO Problem

The proposed objective functions reflecting the considerations mentioned above tend to be competitive with each other. In this study, the minimum deviation method [3], one solution method of MOO, was used to find the preferred compromise solution. The general statement of programming with k objective functions is given as follows:

$$MinimizeF = \sum_{n=1}^{k} w_n \left[\frac{f_n\left(ST_i^*, LT_i^*\right) - f_n(ST_i, LT_i)}{f_n\left(ST_i^*, LT_i^*\right)} \right]$$

where $f_n\left(ST_i^*, LT_i^*\right)$ is the value of objective function n at its individual optimum ST^*andLT^*; $f_n\left(ST_i, LT_i\right)$ is the function itself, and w_n indicates the relative importance that the decision maker attaches to objective function n which must be specified for each of the k objective functions.

4 Computational Experiment

All 248 investigated fields were classified into nine datasets depending on their field size and distance to the factory. The randomly selected fields of 9 datasets were put into the simulation models to calculate the parameters, as described in Section 2. All obtained values for each dataset were used simultaneously to perform SOO and MOO based on the proposed objective functions and the minimum deviation method respectively. In order to compare the outputs of mechanized resource allocation obtained from SOO and MOO, the costs of each working group engaged in these processes were calculated.

5 Results

By using the minimized deterioration time as an objective function, this plan gives sugarcane farmers and sugar factories an advantage

over machinery owners. The machinery owner would have to pay more money for fuel costs due to increases in the total travel distance.
By using minimized the total traveling distance, the difference in the number of working days required to bring the sugarcane to the factory was unacceptable, especially for sugarcane farmers waiting for harvest to occur. This difference would cause them to lose the opportunity to obtain a better price, because their fields would possibly be harvested late or delayed the chance of harvesting the sugarcane when its sugar content is high. The sugar factory would also lose sugar productivity due to this delay.
The total number of trucks allocated by the minimization number of operating days was larger than the total number of trucks allocated by MOO. Thus, by using minimization of the number of operating days sometimes would be inappropriate in regions where the number of available trucks is limited, especially the number of 10-wheeled trucks.

6 Conclusions

6.1 Under the limited mechanized resources of Thai sugarcane harvesting and transportation, the usage of multi-objective optimization (MOO) has demonstrated more proper allocation of mechanized resources to sugarcane fields than single-objective optimization in the aspects of the distribution of operating costs and the operation time.

6.2 When comparing the result of the MOO with the currently used truck allocation plan, cost reduction and efficient operation in the processes have been achieved by applying MOO via truck allocation planning. The percentage of reduction in operating cost was in the range of 4 to 9%. The cost could be possibly reduced to the range of 0.06 to 0.14 US$ per ton (or 2 to 5 baht/ton). The percentage of decrease in the number of working days per unit area was in the range of 4 to 43%.

References

1. Singh, G., Abeygoonawardana, K. Computer simulation of mechanical harvesting and transporting of sugarcane in Thailand. Agr. Sys., 8:105-114, 1982.
2. Rungrat, K., Bootrach, S., Somudorn, C., Nanun, B., Tinnangwattana, T. Study the effect of green and burned cane on sugar process *(in Thai)*. Office of the Cane and Sugar Board, Ministry of Industry, Thailand, 2000.
3. Tapan, P. B. Multiobjective Scheduling by Genetic Algorithms. Kluwer Academic, Boston, US (1999).

Efficiency Measurement of Organizations in Multi-Stage Systems

Andreas Kleine

University of Hohenheim (510 A), Institute of Business Administration, Quantitative Methods, D-70593 Stuttgart. ankleine@uni-hohenheim.de

Summary. Traditional Data Envelopment Analysis (DEA) characterizes decision making units by a vector of external inputs and outputs. By the use of a scalarizing function the inputs and outputs are aggregated to an efficiency measure for each unit. DEA models are based on the assumption that the production process is a "black box", i.e. inputs are transformed in this box into outputs. In many cases more information about the production process is available. This is especially the case in multi-stage production systems. Decision making units of the underlying network employ intermediate and external inputs simultaneously. Unlike external inputs, which are assumed in classical models, intermediate inputs are provided directly by decision making units of the network. This means that intermediate goods affect the performance measure of at least two decision making units, the unit providing services and the unit applying these services. For this very reason a general DEA-model is introduced, which takes the special features of units in a multi-stage system into consideration.

1 Introduction

Leading-edge companies have realized that the real competition is not company against company, but rather supply chain against supply chain [3]. This statement points out that a comparison of companies requires an analysis of the underlying supply chain. If we want to meet the above-mentioned claim, we have to measure the efficiency of multi-stage systems rather than the efficiency of companies on its own.

The Data Envelopment Analysis (DEA) is an approach to compare relative efficiency of decision making units (DMUs) in general. Traditional DEA approaches deal with the production process as a black box [4, 12]. Each DMU_j ($j \in \mathsf{J}$) is characterized by external inputs x^e_{mj} and external outputs y^e_{nj} ($m \in \mathsf{M}^e, n \in \mathsf{N}^e$, cf. Fig. 1). Often, more

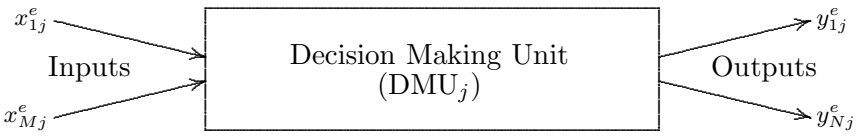


Fig. 1. Inputs and outputs of DMU$_j$

detailed information on the allocation process is available. In addition to classical external (direct) factors we possibly have information on interdependencies. An efficiency analysis has to include these additional intermediate inputs and intermediate outputs.

DMUs of the service sector are often characterized by multi-stage systems. Look for example at a bank with several agencies. All agencies are usually organized in the same manner. Now, the question raises, which agency is the best? Each agency consists of departments which apply intermediate and external factors, e.g. a loan division has employees – external input – and gets information on costumers from a revision department – intermediate input. The quantity and quality of loan decisions depends on both qualification of staff and validity of information. Moreover, these decisions do not only affect the performance of the agency as a whole, but they can influence the performance of other departments as well. A performance measurement should take into consideration these interdependencies.

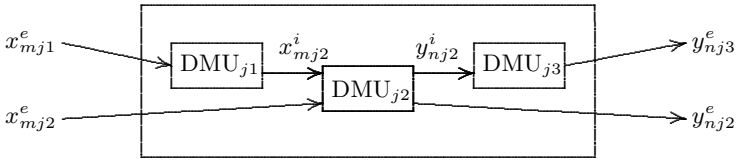


Fig. 2. Interdependent inputs and outputs of sub-DMU$_{j2}$

Figure 2 shows a simple example of a multi-stage system with three sub-organizations: DMU$_{jk}$ $(k = 1, 2, 3)$. On the one hand sub-DMU$_{j2}$ requires external input x^e_{mj2}, on the other hand it applies intermediate inputs x^i_{mj2} which are provided by DMU$_{j1}$. Moreover, DMU$_{j2}$ supplies intermediate outputs $y^i_{nj2}(= x^i_{mj3})$ for DMU$_{j3}$ and external outputs y^e_{nj2} for the market. The efficiency of sub-DMUs is subject of this article.

Attempts to model DMUs that exhibit an intermediate structure are known in literature. Golany, Hackman and Passy [8] construct a DEA approach for particulars multi-stage system, Zhu et al [5, 12] analyze a specific seller-buyer chain incorporating cooperative structures, Färe et al [6, 7] present a dynamic DEA network model. In the following we introduce a linear nonoriented performance model which is based on non-discretionary factors. Section 2 summarizes a global DEA efficiency measure for a whole DMU, section 3 develops a measure for sub-DMUs in multi-stage systems.

2 Global Efficient DMUs

By the help of traditional DEA models we can directly analyze the efficiency of a multi-stage system as a whole. External inputs at the beginning of a chain and external outputs at the end characterize a multi-stage DMU. Often special external inputs and outputs are used for an adequate specification of a supply chain [2, 11], e.g. cycle time, service level etc.

In accordance with figure 1 DMU_j (j=o) is called *global efficient*, if there does not exist an external input-output-combination which is better in at least one external input or output and which is not worse in all other external inputs and outputs. The following model determines the global efficiency of multi-stage DMU_o. In contrast to [10] we use a nonoriented model because it is assumed that external inputs and outputs can be directly influenced by each multi-stage DMU:

$$
\begin{aligned}
&\max\ \theta_o + \epsilon\big(\textstyle\sum_m d_m^{e-} + \sum_n d_n^{e+}\big) \\
&\text{s.t} \\
&\textstyle\sum_j \lambda_j\ x_{mj}^e + d_m^{e-} = x_{mo}^e\ (1-\theta_o) && (m \in \mathsf{M}^e) \\
&\textstyle\sum_j \lambda_j\ y_{nj}^e\ - d_n^{e+} = y_{no}^e\ (1+\theta_o) && (n \in \mathsf{N}^e) \\
&d_m^{e-}, d_n^{e+} \geqq 0 && (m \in \mathsf{M}^e, n \in \mathsf{N}^e) \\
&(\lambda_1, \ldots, \lambda_J)^{\mathsf{T}} \in \Lambda \subseteq \mathbf{R}^J
\end{aligned}
$$

Theorem 1. *A multi-stage DMU_o is global efficient, iff optimal θ_o^*=0 and $d_m^{e-*} = d_n^{e+*} = 0$ $(m \in \mathsf{M}^e, n \in \mathsf{N}^e)$.*

The proof of theorem 1 follows from well-known theorem of classical DEA [9, p.181]. If a multi-stage DMU is inefficient, external inputs and outputs are improvable θ-times. Note that ϵ is a small positive 'non-Archimedean' number in order to guarantee an efficient solution.

Whether a DMU is global efficient depends on inputs and outputs of multi-stage DMUs and the technology assumed in Λ, e.g. constant returns to scale (crs), variable returns to scale (vrs), free disposal hull (fdh) etc.

3 Efficiency of Interdependent DMUs

In the following we investigate the efficiency of sub-DMU$_{jk}$ which is located on stage k of DMU$_j$ ($k \in \mathsf{K} = \{1, \ldots, K\}$). A performance measurement of sub-DMUs requires extra information on the interdependent multi-stage system. It is assumed that the production process of all DMUs is organized in the same way. Each sub-DMU$_{jk}$ at stage k may require intermediate inputs x^i_{mk} ($m \in \mathsf{M}^i_k$) besides external inputs x^e_{mk} ($m \in \mathsf{M}^e_k$). Similarly, intermediate outputs y^i_{nk} ($n \in \mathsf{N}^i_k$) and/or external outputs y^e_{nk} ($n \in \mathsf{N}^e_k$) are provided by each sub-DMU$_{jk}$ (confer fig. 2).

A sub-DMU$_{jk}$ is efficient at stage k of the multi-level system, if there does not exist a better input-output-combination with respect to all intermediate and external factors at stage k.
In the following it is assumed that intermediate inputs and outputs are not directly controllable [1]. These non-discretionary intermediate factors influence the efficiency measure indirectly. Thus we can measure the efficiency of a sub-DMU$_{ok}$ at stage k:

$$\max\ \theta_{ok} + \epsilon\Big(\sum_{m\in\mathsf{M}^e_k} d^{e-}_{mk} + \sum_{m\in\mathsf{M}^i_k} d^{i-}_{mk} + \sum_{n\in\mathsf{N}^e_k} d^{e+}_{nk} + \sum_{n\in\mathsf{N}^i_k} d^{i+}_{nk}\Big)$$

s.t

$$\begin{aligned}
&\textstyle\sum_j \lambda_{jk}\, x^e_{mjk} + d^{e-}_{mk} = x^e_{mok}\,(1-\theta_{ok}) && (m \in \mathsf{M}^e_k, \textit{ext. input})\\
&\textstyle\sum_j \lambda_{jk}\, x^i_{mjk} + d^{i-}_{mk} = x^i_{mok} && (m \in \mathsf{M}^i_k, \textit{int. input})\\
&\textstyle\sum_j \lambda_{jk}\, y^e_{njk} - d^{e+}_{nk} = y^e_{nok}\ (1+\theta_{ok}) && (n \in \mathsf{N}^e_k, \textit{ext. output})\\
&\textstyle\sum_j \lambda_{jk}\, y^i_{njk} - d^{i+}_{nk} = y^i_{nok} && (n \in \mathsf{N}^i_k, \textit{int. output})\\
&d^{e-}_{mk}, d^{i-}_{m'k}, d^{e+}_{nk}, d^{i+}_{n'k} \geqq 0 \quad (m \in \mathsf{M}^e_k, m' \in \mathsf{M}^i_k, n \in \mathsf{N}^e_k, n' \in \mathsf{N}^i_k)\\
&(\lambda_{1k}, \ldots, \lambda_{Jk})^{\mathsf{T}} \in \Lambda \subseteq \mathbf{R}^J
\end{aligned}$$

Theorem 2. *Sub-DMU$_{ok}$ is efficient at stage k, iff optimal $\theta^*_{ok} = 0$ and $d^{e-*}_{mk} = d^{i-*}_{m'k} = d^{e+*}_{nk} = d^{i+*}_{n'k} = 0$ ($m \in \mathsf{M}^e_k, m' \in \mathsf{M}^i_k, n \in \mathsf{N}^e_k, n' \in \mathsf{N}^i_k$).*

If we are interested in the efficiency of a multi-stage system as a whole with a simultaneous consideration of interdependencies between sub-DMUs, a combination of the above models is necessary. We measure

total efficiency of DMU_o over all stages $k = 1, \ldots, K$ by the use of a meta-variable θ'_0. This measure is applied to each stage $k \in \mathsf{K}$:

$$\max\ \theta'_o + \epsilon\Big(\sum_{m\in\mathsf{M}^e_k} d^{e-}_{mk} + \sum_{m\in\mathsf{M}^i_k} d^{i-}_{mk} + \sum_{n\in\mathsf{N}^e_k} d^{e+}_{nk} + \sum_{n\in\mathsf{N}^i_k} d^{i+}_{nk}\Big)$$

s.t

$$\sum_j \lambda_{jk}\, x^e_{mjk} + d^{e-}_{mk} = x^e_{mok}\ (1-\theta'_o) \qquad (m \in \mathsf{M}^e_k, k \in \mathsf{K})$$

$$\sum_j \lambda_{jk}\, x^i_{mjk} + d^{i-}_{mk} = x^i_{mok} \qquad (m \in \mathsf{M}^i_k, k \in \mathsf{K})$$

$$\sum_j \lambda_{jk}\, y^e_{njk} - d^{e+}_{nk} = y^e_{nok}\ (1+\theta'_o) \qquad (n \in \mathsf{N}^e_k,\ k \in \mathsf{K})$$

$$\sum_j \lambda_{jk}\, y^i_{njk} - d^{i+}_{nk} = y^i_{nok} \qquad (n \in \mathsf{N}^i_k,\ k \in \mathsf{K})$$

$$d^{e-}_{mk}, d^{i-}_{m'k}, d^{e+}_{nk}, d^{i+}_{n'k} \geqq 0 \quad (m \in \mathsf{M}^e_k, m' \in \mathsf{M}^i_k, n \in \mathsf{N}^e_k, n' \in \mathsf{N}^i_k, k \in \mathsf{K})$$

$$(\lambda_1, \ldots, \lambda_J)^{\mathsf{T}} \in \Lambda \subseteq \mathbf{R}^J$$

Theorem 3. *The optimal performance measure θ'^*_o of DMU_o corresponds to the minimum of optimal solutions over all stages θ^*_{ok}:* $\theta'^*_o = \min\left\{\theta^*_{o1}, \ldots, \theta^*_{oK}\right\}$.

The proof of this theorem is based on a familiar transformation of a maximin formula into a linear program with an additional variable, here θ'_o [10].

According to theorem 3 it is not necessary to compute the solutions of the recent model. The optimal solution is directly deducible: The best sub-DMU_{ok} determines total efficiency of DMU_o. Hence, theorem 3 points out that a detailed examination of all sub-DMUs is indispensable.

If we do not use fixed intermediate factors, but rather changeable ones – i.e intermediate inputs and outputs are multiplied by an additional variable – the model will correspond to an approach by Zhu [12]. The performance measure θ'_0 is a lower bound of Zhu's model.

4 Conclusion

The introduced approach is based on the assumption that the underlying multi-stage systems are directly comparable, i.e. all sub-DMUs are similarly organized. However, this condition is only limited to sub-DMUs compared, so that we can neglect remaining connections.

The approaches are for example applicable for measuring performance of agencies arranged in the same manner, e.g. agency of car rentals, banks, insurers etc. Moreover, we can detect inefficiencies of specific departments which for instance offers IT services [12]. If the analyze of supply chains is focused on buyer-seller connections, we can

directly apply these approaches as well. In addition we are able to measure efficiency of multi stage systems over several periods of time in order to detect improvements or problems in a particular system. In these cases performance measure of multi-stage DMUs provide helpful information on the efficiency of sub-DMUs and the underlying organizational structure.

References

1. Banker RD, Morey RC (1986) Effidency analysis with exogeneously fixed inputs and outputs, Operartions Research 34: 513–521.
2. Chen IJ, Paulraj A (2004) Understanding supply chain management: critical research and a theoretical framework. International Journal of Productions Research 42: 131–163.
3. Christopher M (1992) Logistics and supply chain Management, Pitman, London.
4. Cooper WW, Seiford LM, Tone K (2005) Data envelopment analysis. Kluwer, Boston.
5. Cook WD, Liang L, Yang F, Zhu J (2007) DEA models for supply chain or multi-stage structure. In Zhu J, Cook WD (eds) Modeling data irregularities and structural complexites in data envelopment analysis. Springer, New York, 189–208.
6. Färe R, Grosskopf S (2000) Network DEA. Socio-Economic Planing Sciences 34: 35–49.
7. Färe R, Grosskopf S, Whittaker G (2007) Network DEA. In Zhu J, Cook WD (eds) Modeling data irregularities and structural complexites in data envelopment analysis. Springer, New York, 209–240.
8. Golany B, Hackman ST, Passy U (2006) An efficiency measurement framework for multi-stage production systems. Annals Operations Research 145: 51–68.
9. Kleine A (2002) DEA-Effizienz. DUV, Wiesbaden.
10. Kleine A (2006) Effizienz von Supply Chains. In Jacquemin M, Pibernik R, Sucky E (eds), Quantitative Methoden der Logistik und des Supply Chain Management. Kovac, Hamburg, 21-39.
11. Ross AD, Droge C (2003) An analysis of operations efficiency in large-scale distribution systems. Journal of Operations Management 21: 673–688.
12. Zhu, J (2003) Quantitative models for performance evaluation and benchmarking. Kluwer, Boston.

Part XV

Production and Service Operations Management

Construction Line Algorithms for the Connection Location-Allocation Problem

Martin Bischoff and Yvonne Bayer*

Institute of Applied Mathematics,
University of Erlangen-Nuremberg, Germany
bischoff@am.uni-erlangen.de

Summary. In the connection location-allocation problem we are given a set of material flows between pairs of existing facilities each of which must be routed through a connection facility. The objective is to minimize the total transportation costs by locating a given number of connections and allocating the flows accordingly.

For this problem many properties and solution methods of the well-known facility location-allocation problem can be transferred, among others the construction line algorithm, an exact solution method based on discretization results under polyhedral gauge distances.

We have implemented construction line algorithms for the connection location-allocation problem without restrictions as well as in the presence of forbidden regions or barriers. We considered various distance functions, ranging from the Manhattan distance to mixed polyhedral gauge distances and applied hull properties to further reduce the dominating set.

1 The Connection Location-Allocation Problem

The *connection location-allocation problem* (CLP) addresses the question of optimally locating bottleneck locations like junctions, transshipment points or storage buildings of a shipping company. It can also be applied for the planning of passages and alleyways between a given set of buildings or installations as, for example, exhibition centers or factories. Possibly, when taking further restrictions like barriers or forbidden regions into account, one may think of applications as the location of entrances of big constructions like stadiums, hospitals, theme parks or nature preserves or, in a larger scale, the location of bridges, border

* This work was partially supported by DFG grant Kl 1076/8-1.

crossings or tunnels. See [8] for further example of real-world applications covered by this location problem.

The (CLP) can formally be described as follows. We are given a set of L *existing facilities* $\mathcal{A} = \{\mathbf{a}_1, \dots, \mathbf{a}_L\}$, $\mathbf{a}_l \in \mathbb{R}^2$, $l = 1, \dots, L$ and a set of M *flows* with index set $\mathcal{M} = \{1, \dots, M\}$. Each flow $m \in \mathcal{M}$ is associated with a source facility $\mathbf{a}_{s_m}$, a target facility $\mathbf{a}_{t_m}$, $s_m, t_m \in \{1, \dots, L\}$ and an intensity $w_m > 0$. We are interested in minimizing the total transportation cost by locating a given number of connection facilities $\mathbf{x}_n \in \mathbb{R}^2$, $n = 1, \dots, N$, and allocating the flows accordingly. The costs of flow $m \in \mathcal{M}$ are given by

$$c_m(\mathbf{x}) = w_m\big(d_{s_m}(\mathbf{a}_{s_m}, \mathbf{x}) + d_{t_m}(\mathbf{x}, \mathbf{a}_{t_m})\big), \quad \forall \mathbf{x} \in \mathbb{R}^2,$$

where $d_l : \mathbb{R}^2 \times \mathbb{R}^2 \to \mathbb{R}$, $l \in \{1, \dots, L\}$, are distance functions in the plane.

The allocation of connection facilities to flows is established by the binary variables y_{mn}, $m = 1, \dots, M$, $n = 1, \dots, N$, where

$$y_{mn} = \begin{cases} 1 & \text{if connection facility } n \text{ is allocated to flow } m, \\ 0 & \text{otherwise.} \end{cases}$$

Since every flow must be allocated to exactly one connection facility, we require that $\sum_{n=1}^{N} y_{mn} = 1$ for all $m \in \mathcal{M}$. Thus, the set of all feasible assignments $Y \in \{0,1\}^{M \times N}$ which satisfy this restriction is given by

$$\mathcal{Y} = \Big\{ Y \in \{0,1\}^{M \times N} : \; Y = (y_{mn})_{\substack{m=1,\dots,M, \\ n=1,\dots,N}}, \; \sum_{n=1}^{N} y_{mn} = 1, \; \forall m \in \mathcal{M} \Big\}.$$

We therefore obtain the following problem formulation for the (CLP):

$$\begin{aligned} \min \; & \sum_{m=1}^{M} \sum_{n=1}^{N} y_{mn} w_m \big(d_{s_m}(\mathbf{a}_{s_m}, \mathbf{x}) + d_{t_m}(\mathbf{x}, \mathbf{a}_{t_m})\big) \\ \text{s.t.} \; & \sum_{n=1}^{N} y_{mn} = 1, \; m = 1, \dots, M \\ & y_{mn} \in \{0,1\}, \; m = 1, \dots, M, \; n = 1, \dots, N \\ & \mathbf{x}_n \in \mathbb{R}^2, \qquad n = 1, \dots, N \end{aligned}$$

This problem has previously been considered in [8, 9, 2, 1]. It bears a strong resemblance to the facility location-allocation problem (FLP), also denoted as multi Weber problem, which was introduced by [3] and nowadays is one of the best-known multi facility location problems. In

contrast to (CLP), where connections are located with respect to flows, the objective of (FLP) is to minimize the sum of weighted distances from new facilities to existing facilities to model, e.g., the location of warehouses. For a detailed survey and further references on (FLP), we refer to the textbook [4].

2 The Construction Line Algorithm

In order to apply the construction line algorithm, the distances in the objective function of (CLP) must be induced by polyhedral gauges, i.e., $d_l(\mathbf{x}, \mathbf{y}) = \gamma_l(\mathbf{y} - \mathbf{x})$.

A gauge $\gamma : \mathbb{R}^2 \to \mathbb{R}$ in the plane with unit ball $\mathcal{S}$ is defined as

$$\gamma(\mathbf{x}) := \inf\{\lambda > 0 : \ \mathbf{x} \in \lambda\mathcal{S}\}, \quad \forall \mathbf{x} \in \mathbb{R}^2,$$

where $\mathcal{S}$ is a compact and convex set in $\mathbb{R}^2$. A polyhedral gauge is a gauge with a polyhedral unit ball which has a finite number of extreme points [13]. If $\mathcal{S}$ additionally is symmetric, then γ is a block norm [14]. Let $\{\mathbf{v}_1, \ldots, \mathbf{v}_I\}$ be the set of extreme points of $\mathcal{S}$. The corresponding vectors $\mathbf{v}_1, \ldots, \mathbf{v}_I$ are denoted as fundamental vectors of γ. The convex cone spanned by the extreme points $\mathbf{v}_{j_k} \in \{\mathbf{v}_1, \ldots, \mathbf{v}_I\}$, $k = 1, \ldots, K_j$, of the same face of $\mathcal{S}$ is denoted as fundamental cone.

Based on the fact that a polyhedral gauge is linear in each of its fundamental cones [5], a tessellation of the plane can be defined with regions in which the Weber objective function with polyhedral gauge distances is linear. Consequently, an optimal solution of the Weber problem consists of subsets which are either grid points, line segments or whole cells of the grid corresponding to this tessellation. Further, there exists a grid point which is an optimal solution of the Weber problem and it is therefore sufficient to restrict the search to the set of grid points.

This grid is known as construction grid and the corresponding discretization result has been transferred to various location problems similar to the Weber problem. For the Weber problem in the presence of forbidden regions or barriers corresponding discretization results have been investigated, see respectively [7] and [12]. Based on the fact that for arbitrary allocations $Y \in \mathcal{Y}$ also the objective function of (CLP) results in a sum of weighted polyhedral gauge distances, [9] specified a construction grid for (CLP) with polyhedral gauges. An equivalent construction grid for (CLP) with mixed polyhedral gauge distances is given by

$$G = \bigcup_{l=1}^{L} \bigcup_{i=1}^{I_l} \{\mathbf{a}_l + \lambda \mathbf{v}_{li},\ \lambda \in \mathbb{R}\},$$

where $\mathbf{v}_{li}$, $i = 1, \ldots, I_l$ denotes the i-th fundamental vector of the polyhedral gauge corresponding to existing facility $\mathbf{a}_m$, $m = 1, \ldots, M$. Accordingly, the construction grid consists of grid lines passing through the existing facilities in the directions of their corresponding fundamental vectors.

The construction line algorithm determines the optimal solution of (CLP) by evaluating all possibilities of locating the connections on the grid points. Note that the number of combinations of N connections out of the set of grid points increases exponentially with the complexity of the problem.

We therefore applied hull properties to reduce the set of grid points to a dominating subset which contains at least one optimal solution. For block norms we applied the metric hull [5], which results in the rectangular hull if the distance function is the Manhattan distance [10]. A description of how the metric hull can be computed for block-norm distances in the plane is given in [6].

Construction Line Algorithm:

1. Compute the set of all grid points.
2. Reduce the number of grid points by applying hull properties.
3. For all combinations of selecting N elements out of the grid points: Evaluate the objective value of that solution where the connections located on the N selected grid points.
4. A solution which yields the smallest objective is a global optimal solution.

Furthermore, we derived construction grids for the (CLP) with mixed distances in the presence of forbidden regions and barriers, see respectively [7] and [12] for their corresponding definitions for the Weber problem. Besides a more sophisticated computation of distances in the presence of barriers which is based on the visibility graph [11], the three main steps of the construction line algorithm as described above remain the same for both model extensions.

If the underlying gauge distance is symmetric, also for these restricted location problems hull properties can be applied. In particular we used the iterative convex hull [11] to restrict the grid points to a dominating subset in the presence of barriers. In case of forbidden regions we applied what we call the extended convex hull, which is the union of the convex hull and the borders of all its intersecting barriers.

3 Numerical Results and Conclusions

We implemented the proposed construction line algorithms for (CLP) in Matlab, Release 14 and used an Intel Core2 Duo, 2x 1.60 MHz with 2048 MB RAM for evaluation. In the following we provide a short overview of the most relevant results.

Obviously, the computation time increases drastically with the number of flows, connection locations, the complexity of the underlying gauge distances and the given restrictions. Under Manhattan metrics without restrictions it took less than five minutes to solve problems with 435 flows and two connections, 105 flows and three connections or 55 flows and four connections. For a relatively small problem instance with 21 flows and three connections to locate, it took only $0.36s$ to determine the optimum solution under the Manhattan metric, whereas for a block norm distance with six fundamental directions the computation time increased to $5.32s$. In the presence of two forbidden regions, the problem was optimally solved in $3.96s$ and $32.89s$ for the Manhattan metric and the block norm, respectively. The computation time for both distance functions increases drastically to $43.87s$ and $1130.20s$ if the forbidden regions are replaced by two barriers. Mixed distances induced by non-symmetric gauges additionally have an impact on the computation time. Consider the difference of $2.18s$ and $33.99s$ for a (CLP) with 45 flows and three connections under uniform Manhattan metrics on the one hand, and mixed polyhedral gauge distances with not more than four unit vectors on the other hand.

The reduction of the continuous solution space to a finite dominating set is useful not only for the construction line algorithm. Once the distances between the existing facilities and the grid points are known, the continuous multi-facility location problem results in a discrete p-median problem. Instead of the total enumeration approach described in this paper, p-median heuristics can be applied after the reduction in order to tackle larger-size problems.

References

1. M. Bischoff and K. Dächert. Allocation search methods for a generalized class of location-allocation problems. European Journal of Operational Research (in press), doi:10.1016/j.ejor.2007.10.022, 2007.
2. M. Bischoff and K. Klamroth. Two branch & bound methods for a generalized class of location-allocation problems. Technical Report 313, Institute of Applied Mathematics, University of Erlangen-Nuremberg, 2007.

3. L. Cooper. Location-allocation problems. *Operations Research*, 11:331 – 343, 1963.
4. Z. Drezner and H.W. Hamacher, editors. *Facility Location: Applications and Theory*. Springer, 2002.
5. R. Durier and C. Michelot. Geometrical properties of the Fermat-Weber problem. *European Journal of Operational Research*, 20:332–343, 1985.
6. R. Durier and C. Michelot. On the set of optimal points to the Weber problem: Further results. *Transportation Science*, 28:141–149, 1994.
7. H.W. Hamacher and K. Klamroth. Planar Weber location problems with barriers and block norms. *Annals of Operations Research*, 96:191–208, 2000.
8. S. Huang. *The Connection Location and Sizing Problem: Models, Methods and Applications to Supply Chain Design*. PhD thesis, The State Univeristy of New York and Buffalo, 2004.
9. S. Huang, R. Batta, K. Klamroth, and R. Nagi. The K-connection location problem in a plane. *Annals of Operations Research*, 136:193–209, 2005.
10. H. Juel and R.F. Love. Hull properties in location problems. *European Journal of Operational Research*, 12:262–265, 1983.
11. K. Klamroth. *Single-Facility Location Problems with Barriers*. Springer, New York, 2002.
12. S. Nickel. *Discretization of Planar Location Problems*. PhD thesis, University of Kaiserslautern, 1995.
13. R.T. Rockafellar. *Convex Analysis*. Princetown University Press, Princetown, N.J., 1970.
14. J.E. Ward and R.E. Wendell. A new norm for measuring distance which yields linear location problems. *Operations Research*, 28:836–844, 1980.

Service-Level Oriented Lot Sizing Under Stochastic Demand

Lars Fischer, Sascha Herpers, and Michael Manitz

Universität zu Köln, Seminar für Allgemeine Betriebswirtschaftslehre,
Supply Chain Management und Produktion, 50923 Köln
lars.fischer@wiso.uni-koeln.de
herpers@wiso.uni-koeln.de
manitz@wiso.uni-koeln.de

Summary. In this paper, we analyze lot sizing under stochastic demand. The lot sizes are determined such that a target service level is met. This optimization procedure requires the calculation of the shortages and their probability distribution considering the inventory dynamics. For an example, we compare different production plans that reveal the influence of a service-level constraint on lot sizing.

1 The Model

In practical applications, stochastic demand is taken into account by overestimating the demand, usually with a certain multiple of the mean forecast error. Having done such a demand data manipulation, a common deterministic lot-sizing algorithm can be used which is a trade-off between setup and holding costs. In addition, under stochastic demand, lot-sizing affects the service that is offered to the customers. This is what we analyze in this paper by integrating both views. For a given setup pattern, the lot sizes are determined such that the service-level requirements are met, i. e. as small as possible to reduce holding costs but subject to a desired service level. Hence, the problem is to find minimal lot sizes for each production period τ that contains the demands up to period t such that a target service level β^* is met:

$$\text{Minimize} \quad q_{\tau t} \quad \text{s. t.} \quad \beta(q_{\tau t}) = 1 - \frac{\mathrm{E}\left\{\sum_{i=\tau}^{t} B_i(q_{\tau t})\right\}}{\mathrm{E}\left\{\sum_{i=\tau}^{t} D_i\right\}} \geq \beta^* \tag{1}$$

The service level β is defined (in terms of expected values) as the portion of immediately delivered units, whereas $1 - \beta$ is the fraction of backorders comparing to the demand over the periods the lot covers. $B_i(q_{\tau t})$ is the amount of backordered demand observed in period i for a given lot size $q_{\tau t}$ ($\tau \in \{1, 2, \ldots, T\}$, $t = \tau, \tau + 1, \ldots$, $i = \tau, \tau + 1, \ldots, t$). D_i denotes the demand in period i which is a generally distributed random variable according to the stochastic demands.

2 Literature Review

In this paper, we analyze the effect of stochastic demands on lot-sizing. There are (roughly) four major areas of research in the literature dealing with that. First, the early works from Bodt et al. [2], Callarman and Hamrin [1], and Wemmerlöv and Whybark [10] investigate the impact of demand uncertainty if common heuristics for the deterministic case are used. Secondly, one can find approaches that take into account the service given to the customers by introducing backorder or shortage costs (for recent examples see Haugen et al. [4], Sox [6]). As opposed to that, other authors use known service-level constraints by employing dynamic variants of order-up-to inventory policies. For pre-specified setup (or replenishment) periods, the lot sizes are determined by the difference of the dynamic target inventory level and the current inventory level (see for example Tempelmeier [9], Tarim and Kingsman [7]). Our contribution belongs to the fourth area of research. One of the earliest works in this field was Silver [5]. The lot sizes are determined such that a target service level is met. For this purpose, it is required to calculate the shortages and their probability distribution considering the inventory dynamics which is shown in the next section.

3 Calculating The Service Level

The amount of backorders per period in Eq. (1) depends on the change in the net inventory. If the net inventory is negative, a stock-out situation occurs. Let I_i be the net inventory in period i ($i = 1, 2, \ldots, T$). Then, $I_i^{\mathrm{B}} = -\min\{I_i, 0\}$ is the amount of shortage in that period. The difference between the shortage at the beginning (after replenishment by production) and at the end (after demand satisfaction) of period i gives the amount of backorders of a particular period:

$$B_i(q_{\tau t}) = I_i^{\mathrm{B,end}} - I_i^{\mathrm{B,prod}} \tag{2}$$

In general, the shortages are an excess of demand over production. Let $D(i)$ denote the cumulative demand, and $q(i)$ the cumulative production quantity up to a certain period i. The initial inventory at the beginning of the first period of the planning horizon may be larger, equal to, or smaller than 0. If it is smaller than 0, then some undelivered demand exists. For initialization, these backorders increase the demand in the first period. On the other hand, if the initial inventory is positive with stock on hand, it has the same effect as an increase of the production quantity in the first period.

For the calculation of the resulting β service level, the expected shortages have to be determined. Given $D(i)$ and $q(i)$, we can calculate these values by using the first-order loss function, $G^1_X(y)$, which describes in general the expected excess over y for a common non-negative random variable X (see for instance Tempelmeier [8], Zipkin [11], Hadley and Whitin [3]). Because replenishments are assumed to be realized at the beginning of a period, i. e. before demand, the shortage after production ($I_i^{\mathrm{B,prod}}$) in period i is the excess of the cumulated demand of $i-1$ time periods over the production quantity during this time plus the actual replenishment of period i:

$$\mathrm{E}\left\{I_i^{\mathrm{B,prod}}\right\} = G^1_{D(i-1)}(q(i)) \qquad (i = 1, 2, \ldots, T) \quad (3)$$

The shortage at the end of a particular period i ($I_i^{\mathrm{B,end}}$) contains the demand of period i. Hence, it follows:

$$\mathrm{E}\left\{I_i^{\mathrm{B,end}}\right\} = G^1_{D(i)}(q(i)) \qquad (i = 1, 2, \ldots, T) \quad (4)$$

Using (2)–(4), the service level according to (1) that can be met up to period t with the lot size $q_{\tau t}$ can be written as:

$$\beta(q_{\tau t}) = 1 - \frac{\sum_{i=\tau}^{t} \left(G^1_{D(i)}(q(i)) - G^1_{D(i-1)}(q(i))\right)}{\mathrm{E}\left\{\sum_{i=\tau}^{t} D_i\right\}} \quad (5)$$

4 Determining The Lot Sizes

As described in Eq. (1), we determine the lot sizes as small as possible such that a given β^* service level is met, $\beta(q_{\tau t}) \geq \beta^*$. We use a very straight-forward iterative procedure to determine the lot sizes for all

production periods given by a particular setup pattern. After having initialized the lot sizes as small as possible, an iterative procedure starts until the lot sizes remain unchanged. For a current lot size $q_{\tau t}$, the achievable service level $\beta(q_{\tau t})$ is calculated. In a number of cases, there is a positive difference between β^* and $\beta(q_{\tau t})$ which signifies that the lot size is too small to meet the service requirements. In that case, with respect to the expected total demand during the periods the lot covers, the lot size $q_{\tau t}$ is increased by an appropriate amount.

Let us consider an example with 1 product, 4 periods, stochastic demands that are normally distributed with expected forecasts of 20, 80, 160, and 85 units, and a standardized coefficient of variation of 10 % for the periods under consideration. All the lots are determined to be minimal such that a target service level of $\beta^* = 0.95$ is met. For the particular setup pattern that suggests a production in $\tau = 1, 3, 4$ (i. e. the first lot covers the first two periods), the optimal lot sizes are: $q_{12} = 97.01$, $q_{33} = 161.41$, and $q_{44} = 95.55$.

5 Numerical Experiments

The stochastics become evident by the fact that the demands will not exactly be matched with the forecasts. In fact, a certain forecast error E is inevitable due to the stochastic nature of demand. Based on describing these forecast errors, the stochastic demand is modeled in this paper.

To reveal the effects of lot sizing under a service-level constraint, we compare two different situations: (I) production quantities equal to the β^*-fraction of the demand per period, and (II) — in the sense of Section 4 — optimal production quantities such that a target service level β^* is met with minimal lot sizes over the planning horizon. This means, lot sizing with a type-I policy is emulated by simply summing-up the production quantities according to a specified setup pattern. For a lot-for-lot production, both lot-sizing policies are identical.

In Tab. 1 and 2, the results for different stationary forecast-error distributions (represented by a standardized mean $\mu_E = 1$ and varying standard deviation σ_E) and different target service levels β^* are shown. The right half-sides of the tables show the results of type-II lot-sizing policy by which the production quantities are optimized in the sense of Eq. (1). For the same setup patterns, the results for the type-I lot-sizing policy are depicted. This policy simply sums up the production quantities that are a β^*-fraction of period demands. This is shown on the left-hand sides of the tables.

For comparison reasons, there is always a production scheduled in period 1, and, therefore, the lot sizes in this period can be compared for both policies. The column entitled Δ^q shows the relative change in the lot size for the first period by the optimization procedure mentioned above (type-II policy) in comparison with the lot sizes according to the summing-up policy (type-I). The column $\beta(q_{1*})$ shows the service level as is achieved by the first lot for the summing-up policy. With optimized lot sizes, the target service level is realized exactly for every service pattern, i. e. in every row of the tables.

Table 1. Lot-sizing results: $\beta^* = 0.95$, $E \sim \text{Normal}(\mu_E = 1, \sigma_E = 0.1)$

first lot	type-I policy					type-II policy				Δ^q
up to $t = *$	$t = 1$	$t = 2$	$t = 3$	$t = 4$	$\beta(q_{1*})$	$t = 1$	$t = 2$	$t = 3$	$t = 4$	[%]
1	19.62	79.04	159.75	95.55	0.950	19.62	79.04	159.75	95.55	0.0
2	98.67	—	159.75	95.55	0.960	97.01	—	161.41	95.55	1.7
3	258.42	—	—	95.55	0.969	250.37	—	—	103.60	3.1
4	353.97	—	—	—	0.988	330.45	—	—	—	6.6

Table 2. Lot-sizing results: $\beta^* = 0.98$, $E \sim \text{Gamma}(\mu_E = 1, \sigma_E = 0.3)$

first lot	type-I policy					type-II policy				Δ^q
up to $t = *$	$t = 1$	$t = 2$	$t = 3$	$t = 4$	$\beta(q_{1*})$	$t = 1$	$t = 2$	$t = 3$	$t = 4$	[%]
1	27.79	103.98	199.01	114.21	0.980	27.79	103.98	199.01	114.21	0.0
2	131.76	—	199.01	114.21	0.984	128.34	—	202.43	114.21	2.6
3	330.77	—	—	114.21	0.988	314.89	—	—	130.09	4.8
4	444.98	—	—	—	0.995	398.02	—	—	—	10.6

6 Observations and Insights

As one can see from Tab. 1 and 2, the Δ^q values are always non-negative which means that the lot sizes can be reduced by using a service-level oriented optimization procedure. In addition, this reduction intensifies with an increasing number of periods a given lot covers, with the variance of the demands, and with the target service-level.

For any particular lot, the service level is exactly met at the end of the time interval the lot covers. This holds also for the last period of the planning horizon. Hence, both of the lot-sizing policies that are considered in this paper ensure a certain β^* service level in the sense that the β^*-fraction of the expected demand over the planning horizon is to be produced. That means, the overall production quantity of the planning horizon is the same. Therefore, the optimization of lot sizes

is a re-allocation of production quantities. Lots that cover more than one period are reduced. The size of the next lot (which is always a single-period demand lot in the examples above) is increased by the same amount.

The more periods a lot covers, the more its size can be reduced due to the risk diversification effect of accumulated demands. Period demands below average compensate for demands above average. A shortage situation occurs only towards the end of a lot cycle, therefore, the larger the lot sizes, the more risk absorbed. This explains why the lot-sizing effects increase with the number of periods a lot covers and with the stochastics of the demand.

References

1. Callarman, T. E., and R. S. Hamrin (1984). A comparison of dynamic lot-sizing rules for use in a single stage mrp system with demand uncertainty. *International Journal of Operations and Production Management 4*(2), 39–48.
2. de Bodt, M. A., L. N. van Wassenhove, and L. F. Gelders (1982). Lot sizing and safety stock decisions in an mrp system with demand uncertainty. *Engineering Costs and Production Economics 6*(1), 67–75.
3. Hadley, G., and T. M. Whitin (1963). *Analysis of Inventory Systems*. Englewood Cliffs: Prentice-Hall.
4. Haugen, K. K., A. Løkketangen, and D. L. Woodruff (2001). Progressive hedging as a meta-heuristic applied to stochastic lot-sizing. *European Journal of Operational Research 132*(1), 116–122.
5. Silver, E. A. (1978). Inventory control under a probabilistic time-varying, demand pattern. *AIIE Transactions 10*(4), 371–379.
6. Sox, C. R. (1997). Dynamic lot sizing with random demand and non-stationary costs. *Operations Research Letters 20*(4), 155–164.
7. Tarim, S., and B. Kingsman (2006). Modelling and computing (r^n, s^n) policies for inventory systems with non-stationary stochastic demand. *European Journal of Operational Research 174*(1), 581–599.
8. Tempelmeier, H. (2006). *Inventory Management in Supply Networks — Problems, Models, Solutions*. Norderstedt: Books on Demand.
9. Tempelmeier, H. (2007). On the stochastic uncapacitated dynamic single-item lotsizing problem with service level constraints. *European Journal of Operational Research 181*(1), 184–194.
10. Wemmerlöv, U., and D. Whybark (1984). Lot-sizing under uncertainty in rolling schedule environment. *International Journal of Production Research 22*(3), 467–484.
11. Zipkin, P. H. (2000). *Foundations of Inventory Management*. Boston et al.: McGraw-Hill.

Real-Time Destination-Call Elevator Group Control on Embedded Microcontrollers

Benjamin Hiller* and Andreas Tuchscherer**

Zuse Institute Berlin, Takustraße 7, D–14195 Berlin, Germany
{hiller,tuchscherer}@zib.de

Summary. We introduce new elevator group control algorithms that are real-time compliant on embedded microcontrollers. The algorithms operate a group of elevators in a destination call system, i. e., passengers specify the destination floor instead of the travel direction only. The aim is to achieve small waiting and travel times for the passengers. We provide evidence, using simulation, that the algorithms offer good performance. One of our algorithms has been implemented by our industry partner and is used in real-world systems.

1 Introduction

Algorithmic control of elevator systems has been studied for a long time. A suitable control should achieve small average and maximal waiting and travel times for the passengers. The waiting time / travel time of a passenger is the time span between the release of the call and the arrival of the serving elevator at the start / destination floor.

Recently, the paradigm of destination call elevator control emerged. In destination call systems, a passenger enters the destination floor (and possibly the number of passengers traveling to this floor). Such a destination call system is very interesting from an optimization point of view, since more information is available earlier, which should allow improved planning.

In this paper we report on elevator control algorithms designed for Kollmorgen Steuerungstechnik, our partner from industry. The algorithm designed is supposed to run on embedded microcontrollers with

* Supported by the DFG research group "Algorithms, Structure, Randomness" (Grant number GR 883/10-3, GR 883/10-4).

** Supported by the DFG Research Center MATHEON *Mathematics for key technologies* in Berlin.

computation times of at most 200 ms using not more than 200 kB of memory. Thus computational resources are very scarce.

Related Work Many elevator control algorithms have been proposed, but only few of them seem to be used in practical systems. Moreover, there is not much literature on destination call systems yet. Tanaka et al. [5] propose a Branch&Bound algorithm for controlling a single elevator, which uses too much computation time to be implemented on an embedded microcontroller. Friese and Rambau [3] developed an algorithm for cargo elevator group control with capacity one based on Integer Programming. Although the algorithm runs in real-time on a PC, it is still too time-consuming for embedded microcontrollers. The book of Barney [1] deals mainly with engineering aspects.

Contribution We introduce new destination call control algorithms suited to run on an embedded system offering very scarce computing resources. Since exact optimization is not feasible on such hardware, the algorithmic approach is an insertion heuristic using a non-trivial data structure to maintain an elevator tour. We assess the performance of our algorithms by simulation. We also compare to algorithms for a conventional system and a more idealized destination call system. This gives an indication of the relative potentials of these systems.

2 Modeling the Destination Call System

The real-world destination call system envisioned by Kollmorgen works as follows. Upon arrival, a passenger enters his destination floor (issues a *call*) and is immediately assigned to one of the elevators of the group. The passenger is supposed to go to the corresponding elevator and board it as soon as it arrives and indicates the correct leaving direction. If the designated elevator arrives and the cabin is full so that a passenger cannot enter, he is supposed to reissue his call.

The anticipated operations of the elevator group can be described by tours for each elevator, specifying the order floors are visited. These tours are used to predict the waiting and traveling times of the passengers, thus allowing to evaluate different control alternatives.

The tours have to fulfill some requirements modeling the real system. (a) Tours need to respect the assignments fixed so far. This requirement differs from the assumptions of Tanaka et al. since there the assignment is done on arrival of an elevator at a floor. (b) A tour must not contain a turn for a passenger, i. e., a passenger must never move in the wrong direction. (c) For each floor and leaving direction,

we assume that *all* passengers assigned to an elevator enter the cabin at its first stop at this floor with the corresponding leaving direction. The rationale for this rule is the following. The elevator control has no way to detect which passengers enter at a certain floor (there are no panels in the cabin). Therefore it does not know which stops are really required by the loaded passengers and the elevator has to stop at all registered destination floors. In fact, this is equivalent to assuming that all waiting passengers enter the elevator and thus the capacity of the elevator is ignored for the planning.

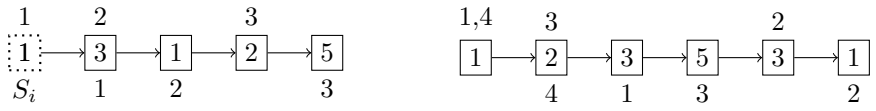
We make some other reasonable assumptions as discussed by Tanaka et al. [5], e. g., that if no passenger has to be served by an elevator, the elevator stays at its last floor and the cabin cannot stop or reverse direction halfway between floors.

Note that due to requirement (c) there may be *phantom stops*, i. e., stops for dropping a passenger who is not really in the cabin. Phantom stops and the immediate assignment are features which might alleviate the advantages of a destination call system since both restrict the optimization potential.

3 Algorithms

The results of Tanaka et al. [5] and Friese and Rambau [3] suggest that it pays off to use thorough optimization for computing new schedules. However, the scarce computing resources available on embedded microcontrollers make exact optimization methods infeasible. We therefore propose insertion heuristics, which are well-known for e. g., the Traveling Salesman Problem. The structure of a tour for an elevator is much more complex, making the insertion operation particularly non-trivial.

A tour T is a list of stops $T = (S_0, \ldots, S_k)$, where each stop is described by its halting floor, its scheduled arrival and leaving times, and the sets of calls picked up and dropped at this floor. Moreover, we also store the set of currently loaded calls (after dropping and picking up) at each stop. A new call c can be inserted into an existing tour T via the operation $\textsc{AddCall}(T, S_i, c)$, where S_i indicates the insertion position. If the floor of S_i does not match the start floor of call c, a new stop is created before S_i. The insertion position for the drop stop is uniquely determined by S_i, the remainder of T, and the no-turn requirement. It may be necessary to split an existing stop into two new stops to avoid direction changes for passengers. Of course, not every choice of S_i is feasible for insertion but it has to be ensured that a feasible tour is obtained afterwards.



(a) Original tour. (b) Tour after insertion.

Fig. 1. Inserting call $4\colon 1 \to 2$ at S_i needs repair. A square represents a stop at a floor. Numbers above/below a stop indicate calls picked up/dropped

AddCall is non-trivial due to the cases that may arise. For instance, consider the tour in Figure 1(a) and suppose we want to insert the new call $4\colon 1 \to 2$ at the first stop at floor 1 (which is the only feasible insertion position). We need to create a new stop at floor 2, leaving upwards due to call 1. But then call 3 will enter and we need to go to floor 5 before we can leave floor 3 downwards. Therefore we need to adjust the tour, keeping it close to the original one. There are more complex cases of this *repair* operation to take into account.

We use the insertion procedure AddCall to set up a group elevator algorithm Best Insertion (BI) as follows. Once a new call enters the system, BI inserts the new call at all feasible positions in the already scheduled tours. The call is assigned to the elevator and insertion position with minimum cost increase. The cost function captures waiting and travel times for all calls. This way, the algorithm balances stability of the plans for old calls with good service for new calls.

In order to achieve real-time compliance and to avoid deferment of single calls leading to high maximum waiting times, we selected a suitable subset of insertion positions. The algorithm CBI (controlled BI) eventually implemented by our industry partner works like BI, but using just this restricted subset of insertion positions.

4 Evaluation and Computational Results

We now use simulation to evaluate our algorithms and compare them to algorithms for a conventional system and a more idealized destination call system. To measure the quality of a control algorithm we use quantiles. We look at the 50%, 75%, 90%, and 100% quantiles of the waiting and travel times.

Simulation Model and Instances The precise rules of the simulation are as follows. At each stop of an elevator the passengers enter the cabin in first-come-first-served (FCFS) manner. Of course this is only relevant if the cabin capacity does not suffice to pick up all waiting passengers.

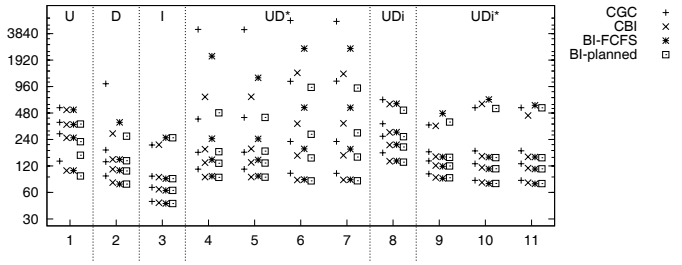
We consider a building with an elevator group serving 16 floors. The passenger data used in our experiments came from the software tool *Elevate* [2]. We look at eleven templates defined by Elevate representing different traffic patterns. These include up traffic (U), down traffic (D), and interfloor traffic (I), as well as combinations of different traffic patterns, e. g., UDi denotes a situation with up and down traffic and a little interfloor traffic. Instances with changing predominant type of traffic are indicated by a "*". For each template we compute the quantiles over ten samples. A more extensive evaluation can be found in [4].

Real Destination Call System For the destination call system described in Section 2, we compared our algorithms BI and CBI to an adapted version of the Kollmorgen algorithm for conventional systems, which is based on collective control [1]. The criterion for assigning calls to elevators aims to minimize the required waiting time. Our results show that BI seems to be superior to the straightforward adaption of the Kollmorgen algorithm, in particular for the travel times and the higher quantiles of the waiting times.

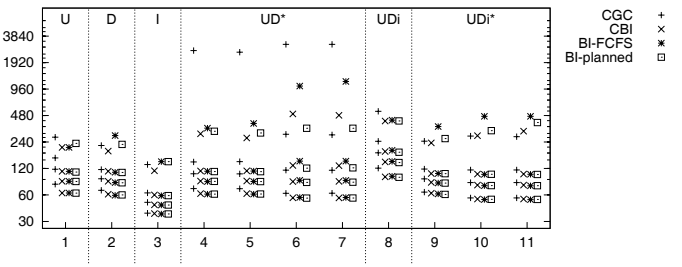
System Comparison For comparison, we also studied two different elevator control systems. In the conventional system we have no information about the destination floor of a call until the passenger has entered the cabin. On the other hand, it is not necessary to assign each call to an elevator immediately. Passengers again enter a cabin in FCFS order. For this setting we implemented an algorithm called CGC [1] designed to perform well in most traffic situations.

Moreover, we consider an idealized destination call system. Here we have complete boarding control, i. e., at each stop of an elevator the control algorithm determines which passengers are picked up. Consequently, the capacity of the elevator cabin is taken into account in this model. We compared BI to another adapted variant of the Kollmorgen algorithm. Similar to the real destination call system, BI performs better in most situations. Finally, we compare the best algorithms of the different systems with each other: CGC (conventional system), CBI and BI-FCFS (real destination call system), and BI-planned (idealized destination call system).

The results for travel times are given in Figures 2(a) and 2(b). BI-planned outperforms the other algorithms with three elevators, while CGC achieves the worst travel time quantiles. CBI performs similarly to BI-FCFS, but CBI always achieves a smaller maximal travel time. Using four elevators, CBI yields similar results as BI-planned and gives an even better maximal travel time on most instances. CGC performs quite well on the I and UDi* instances.



(a) travel time, 3 elevators



(b) travel time, 4 elevators

Fig. 2. Travel times quantiles (in seconds) for the eleven call templates

We summarize the most important results. CBI is almost as good as BI-FCFS and even better for maximal waiting and travel times. Destination call systems seem to be superior to conventional systems in high load situations, the opposite seems to hold for low load at least for the waiting times. Moreover, control about the passengers entering the cabin at a stop pays off for destination call systems.

References

1. G. C. Barney. *Elevator Traffic Handbook: Theory and Practice*. Taylor and Francis, 2002.
2. Elevate – dynamic elevator simulation. Version 6.01. Peters Research, 2004. `www.peters-research.com`.
3. P. Friese and J. Rambau. Online-optimization of a multi-elevator transport system with reoptimization algorithms based on set-partitioning models. *Discrete Appl. Math.*, 154(13):1908–1931, 2006.
4. B. Hiller and A. Tuchscherer. Real-time destination-call elevator group control on embedded microcontrollers. ZIB Report 07-26, , 2007.
5. S. Tanaka, Y. Uraguchi, and M. Araki. Dynamic optimization of the operation of single-car elevator systems with destination hall call registration: Parts I and II. *European J. Oper. Res.*, 167(2):550–587, 2005.

Integrated Design of Industrial Product Service Systems

Henry O. Otte, Alexander Richter, and Marion Steven

Chair of Production Management, Ruhr-University Bochum, Universitätsstraße 150, 44780 Bochum. marion.steven@rub.de*

Summary. In recent years Industrial Product Service Systems (IPSS), characterized by an integrated supply of products and services, have emerged as new business models. The aim of this paper is to compare the traditional business model for simple transactions and the full-service business model. The question of whether the full service business model can contribute to the degree of vertical integration is supposed to be answered.

1 Introduction

Integrating the supply of products and services is accompanied by a change from a transaction- to a relationship-oriented business connection.[6] As a consequence new, long-term business models have emerged in the plant and machinery industry in form of IPSS by incorporating components of products and services. Typical for such business models is an increased level of interaction between the manufacturer and the customer due to the higher degree of service components. The question arises to what extent such business models can contribute to the optimization of the degree of vertical integration.

We compare two institutional arrangements: the traditional business model 'make or buy' and the full-service business model. Regarding the full-service business model, we consider a manufacturer who is responsible for both developing and assuring the operational readiness of a bundle of technological infrastructure.

The paper is structured as follows. In section 2 we introduce the (formal) model. Section 3 compares the business models and section 4 concludes.

* Financial support from the German Science Foundation (DFG) through SFB/TR29 is gratefully acknowledged.

2 Model Description

We consider a business relationship between a customer and a manufacturer of a machine. The customer has to face the decision whether he wants to delegate all activities which lead to operational readiness to the manufacturer or if he wants to purchase the product alone. As depicted in the timeline in figure 1, the analysis is restricted to two periods, the development phase and the operating phase.

The business relationship starts with a contract in $t = 0$. The contract specifies the ownership structure, the basic characteristics of the product (service system) to be delivered and the payment P_0 to be made by the customer. It is assumed that the specified characteristics are observable and verifiable.

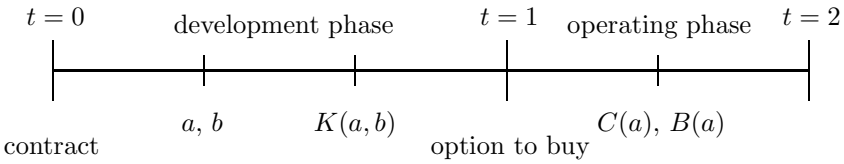


Fig. 1. Timeline

Prior to the production and delivery of the machine, the manufacturer is offered the opportunity to modify the product service system by choosing specific investments, without violating the contract. These non-contractible investments in innovations determine the productivity of the industrial product service system. However, it is important to distinguish between productivity of products and services. For products the correlation of productivity and customer satisfaction (perceived quality) can be assumed to be positive, whereas for customized services it is negative.[2] For instance, on the one hand the manufacturer is able to increase the productivity of the service components by introducing remote services like remote repair, remote diagnosis and remote maintenance. On the other hand, the customer's desire for face-to-face contact tends to result in poor acceptance of such technologies. In addition security concerns arise and can lead to potential barriers due to the customer's fear of having a transparent production.

Therefore, we model a two dimensional investment decision in accordance to [5]. The levels and costs of these two investments are de-

[2] See [1] for an empirical study on this proposition.

noted by a and b respectively. In particular, investment b describes a quality innovation of the machine, whereas investment a reflects the service innovation. Both investments are independent and influence the value of the machine for the customer, $V(a,b)$, and for the manufacturer, $R(a,b)$. It is assumed that the value functions $R(a,b)$ and $V(a,b)$ are concave and increasing in both arguments. For simplicity we also assume that $V_a(a,b) = R_a(a,b)$ and $V_b(a,b) = R_b(a,b)$.[3] Moreover, investment a in service innovation is accompanied by a reduction of customer's benefit from consuming the service, $B'(a) < 0$. Whether an innovation is implemented or not, depends on the approval of the owner. Thus, the allocation of ownership is simply the allocation of control rights. After construction, the costs of building the machine $K(a,b)$ incur, where $K_a(a,b) = K_b(a,b) = 1$ and $K(0) > 0$.

When the initial ownership is assigned to the manufacturer, the customer has the option to buy the improved machine in $t = 1$. It is assumed that the machine has greater value for the customer than for the manufacturer, that is $V(a,b) > R(a,b)$. For reasons of efficiency, the customer always buys the machine. Hence, investment a can be interpreted as hybrid and investment b as cooperative in the sense of [3].

Finally, in the operating phase not only the customer's benefit, $B(a)$, but also the costs of service delivery, $C(a)$, are realized. The cost and benefit functions are supposed to be convex and decreasing in a. Furthermore, we assume that $C(0) > 0$ and $B(0) > 0$. In order to make the implementation of the service innovation efficient ex post, we suppose the cost reduction to be greater than the benefit reduction.

Both parties are risk neutral. The first-best solution (a^*,b^*) is then characterized by the following first order conditions:

$$V_a(a^*, b) - C'(a^*) + B'(a^*) = 1 \tag{1}$$

$$V_b(a, b^*) = 1 \tag{2}$$

We now consider the second-best with non-verifiable investments.

3 Comparison of the Two Business Models

Firstly, we will analyze the problem of bundling goods and services for given ownership structures. In a following step the ownership structures are compared.[4]

[3] In what follows, subscripts are used to denote partial derivatives.

[4] See for such a procedure [2] who analyze the bundling of two tasks in an incomplete contracts framework, too.

3.1 Manufacturer Ownership

As a result of assigning the ownership to the customer ex post, the surplus, $S(a,b) = V(a,b) - R(a,b)$, is supposed to be divided according to the bargaining power, which is α for the manufacturer and $(1-\alpha)$ for the customer.[5] The customer's payment to the manufacturer after renegotiation will then be:

$$P(a,b) = R(a,b) + \alpha S(a,b)$$

In case of a full-service business model the manufacturer maximizes the sum of the machine price $P(a,b)$ and the ex ante determined payoff P_0 less the costs $C(a,b)$ and $K(a,b)$. The profit maximizing choice $a = a^{fs}$ and $b = b^{fs}$ satisfies the following first order conditions:

$$(1-\alpha)R_a(a^{fs},b) + \alpha V_a(a^{fs},b) - C'(a^{fs}) = 1 \quad (3)$$

$$(1-\alpha)R_b(a,b^{fs}) + \alpha V_b(a,b^{fs}) = 1 \quad (4)$$

By contrast, in case of traditional business model the choice is $a = a^{tm}$ and $b = b^{tm}$ to solve:

$$(1-\alpha)R_a(a^{tm},b) + \alpha V_a(a^{tm},b) = 1 \quad (5)$$

$$(1-\alpha)R_b(a,b^{tm}) + \alpha V_b(a,b^{tm}) = 1 \quad (6)$$

When comparing (4) and (6) it becomes obvious that in both business models b equals the reference solution in (2). Moreover, concerning a in both business models the reduced customer benefit ($B'(a) < 0$) is not internalized by the manufacturer. Hence, an overinvestment follows in (3), whereas (5) results in an underinvestment due to the disregard of $C'(a)$. However, it cannot be answered in general, which of the business models is best. For instance, if the benefit reduction is negligible, so is the overinvestment. Thus, the full-service business model would be preferred. This is especially the case for $|2B'(a)| \leq |C'(a)|$. Then, given our assumptions, the level of the corresponding underinvestment $|B'(a^{tm})|$ will be greater than the level of the overinvestment $|-B'(a^{fs})|$.

3.2 Customer Ownership

To implement the innovation, now the approval of the customer is required. Therefore, renegotiation occurs concerning the surplus of an implemented innovation.[5] This implies that the adverse effect of the cost reducing investment b is in part internalized. Furthermore, the

[5] See [4] for non-renegotiation proof option contracts.

highest possible profit is independent of the business model. Specifically, from now on we only refer to a full-service business model when the manufacturer possesses the initial ownership.

The manufacturer sets $a = a^{co}$ and $b = b^{co}$ to solve the first order conditions:

$$\alpha[V_a(a^{co}, b) + B'(a^{co}) - C'(a^{co})] = 1 \tag{7}$$

$$\alpha V_b(a, b^{co}) = 1 \tag{8}$$

A comparison with the first-best solution reveals that the manufacturer underinvests in both components, since he only receives an α-fraction of his investment's contribution to surplus.

3.3 The Choice of Ownership Structure

To restrict the analysis to the full-service business model, we assume $|C'(a)| \geq |2B'(a)|$. The investment levels under the different ownership structures can then be ranked as follows:

$$a^{co} \leq a^* < a^{fs}$$
$$b^{co} \leq b^{fs} = b^*$$

In case of full-service the investment level concerning b matches the first-best solution and is greater than the one in case of customer ownership unless $\alpha = 1$. More difficult to evaluate are the investment incentives concerning a. On the one hand there is overinvestment when the manufacturer initially owns the machine, while on the other hand underinvestment may occurs. We find that the degree of overinvestment is of lesser importance than the degree of underinvestment, if the following condition holds:

$$\frac{\alpha}{1-\alpha} < \left| \frac{V_b(a, b)}{B'(b)} \right| \tag{9}$$

The right-hand side of (9) compares the costs and benefits of a service innovation to the customer. If the machine value, which has been increased by the service innovation, is greater than the benefit reduction, the right-hand side will always be greater than one. In this case the condition holds for a sufficiently small bargaining power $\alpha \leq \frac{1}{2}$. Notice, if $\alpha = 1$, the traditional business model will always be superior unless the negative benefit effect does not apply. However, then in both business models the first-best solution is achieved.

4 Concluding Remarks

In this paper, we consider the supply of IPSS as a long term development and service contract between a manufacturer and a customer. We discuss the effect of the special design of IPSS on the incentive structure of the inter-firm relationship. Relative to the traditional business model, the full-service model as a specific form of IPSS only creates additional value, when the initial ownership is assigned to the manufacturer. Moreover, we were able to show that the advantage of the full-service model is dependent on both the costs and benefits of a service innovation as well as the bargaining power of the manufacturer.

With regard to the quality innovation, IPSS are always superior. The service innovation, however, results in a negative externality. Because of non-internalizing this externality, IPSS turn out to be superior only when the customer's bargaining power is relatively great. The reason is quite simple and stems from the hold-up problem in case of traditional procurement. In such a situation the possible opportunistic behavior of the customer would be more disruptive than the adverse effect of the service innovation.

To counteract the negative externality an integrated design, characterized by interactions between product and service components, is necessary. Future research should therefore take into account a more complex cost-effort function in which a and b are substitutes.

References

1. Anderson W. E., Fornell C., Rust R. T. (1997) Customer satisfaction, productivity, and profitability: Differences between goods and services. Marketing Science 19: 129–145
2. Bennet J., Iossa E. (2006) Building and managing facilities for public services. Journal of Public Economics 90: 2143–2160
3. Che Y.-K., Hausch D. B. (1999) Cooperative damages investments and the value of contracting. American Economic Review 89: 125–147
4. Edlin A. S., Hermalin B. E. (2000) Contract renegotiation and options in agency problems. Journal of Law, Economics and Organization 16: 395–423
5. Hart, O., Shleifer A., Vishny, R. W. (1997) The proper scope of government: theory and an application to prisons. Quarterly Journal of Economics 112: 119–1158
6. Oliva R., Kallenberg R. (2003) Managing the transition from products to services. International Journal of Service Industry Management 14:160–172

Lot Sizing Policies for Remanufacturing Systems

Tobias Schulz

Faculty of Economics and Management, Otto-von-Guericke University Magdeburg. tobias.schulz@ovgu.de

1 Introduction

In recent years many companies have extended the focus of their logistic processes to a closed-loop supply chain perspective in order to meet the growing environmental awareness of their customers. To analyze such systems theoretically the research field of Reverse Logistics has been established which adds to the traditional view of logistics the flow of goods coming back from the customers to the company. After receiving the products back the company can choose one from the manifold options on how to handle these products. Among others, remanufacturing has been proven to be a worthwhile option in several businesses. The process of remanufacturing begins with the disassembly of the returned products. The obtained components are afterwards cleaned and reworked until a good-as-new quality is assured. Finally, when meeting the required quality standards those components can be used in the final product assembly as a low-cost alternative to newly manufactured ones. Thus, a part of the embedded economic value of the returned products can be saved by remanufacturing. While analyzing the process of remanufacturing, a multi-level inventory system has to be evaluated in order to gain insight into the process. Since fixed as well as holding costs prevail on each system level a multi-level lot sizing problem needs to be solved which shall represent the main focus of this contribution.

The remaining paper is structured as follows. While the next chapter describes the problem setting and introduces a heuristic on how this problem can be solved, chapter 3 adapts this heuristic in a way that it finds a good solution for any parameter setting. The last chapter

summarizes the results of this contribution and gives an outlook to future research directions.

2 Problem Setting and Model Formulation

A company engaged in the area of remanufacturing that remanufactures several old products (e.g. engines) coming back from their customers shall be the background for the problem setting. To keep the analysis simple the focus shall be restricted on only one specific remanufactured product named A. Figure 1 presents the general structure of this simplified system. Further simplifications are made regarding the fact that all input data is assumed to be deterministic and that there are neither lead nor processing times.

The demand for the final product A is assumed to be constant and continuous with the rate of λ units per period. In order to satisfy that demand the company manufactures the final product solely by using the only required component B. As there are no considerable fixed costs for the assembly process the final product is assembled with the smallest possible lot size of one and is delivered immediately to the customers. When the customers have no further use for their product A (e.g. it is broken or its leasing contract ends) they have the opportunity to return the product to the company. For the sake of simplicity, these returns which are denoted A' are also assumed to be constant and continuous at which α represents the percentage of old products returned to the company. By disassembling A' the worn component B' is obtained. Although the process of disassembly typically consists of manual work fixed costs prevail for setting up required disassembly tools and/or measuring devices that allow an improved evaluation for the reusability of components before disassembly. Within this model K_d represents the fixed costs for a disassembly batch while h_d is the holding cost incurred for storing one old product for one period. Due to different stages of

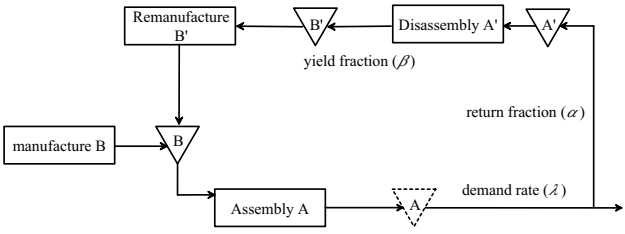


Fig. 1. Material flows in a manufacturing/remanufacturing system

wear not all returned products contain a reworkable component B'. The fraction of reworkable components B' with respect to the total demand rate is denoted by β. As the remanufacturing process incurs fixed costs of K_r for setting up the cleaning and mechanical rework tools a batching of reworkable components takes place as well. Hence, some reworkable components need to be stored resulting in costs of h_r per unit and period. After successfully reworking the components B' they become a good-as-new component B which is held in a serviceables inventory for h_s per unit and period. In order to secure the final product assembly of A some components of B have to be manufactured in addition(as β is commonly smaller than one). The decision relevant fixed costs are denoted by K_m representing the cost for setting up a production lot for component B. Newly manufactured components are held in the same serviceables inventory as remanufactured ones and it is assumed that the holding costs do not differ between both sourcing options. In general, the holding costs (when interpreted as costs for capital lockup) of all levels are connected by the following inequality since more value is added to the component on each level: $h_d < h_r < h_s$.

In a recent work, Teunter et al. [2] have analyzed a quite similar system by a mixed-integer linear program (MILP). Despite the fact that their system does not contain the stage of disassembly and includes the possibility of variable demands and returns the introduced MILP can also be applied to the problem description above. Nevertheless, Teunter et al. conjecture that finding an optimal solution to their problem is NP-hard and adapt therefore well-known simple dynamic lot sizing heuristics. The performance of those heuristics in a stationary demand environment appears to be rather poor (around 8% at average) which motivates the search for a distinct heuristic approach.

Ferretti and Schulz [1] present a common cycle approach which bases on a multi-level EOQ-type analysis to control the system described above in a heuristic manner. In their contribution, three different parameters are identified which are needed to control the system. The first parameter T describes the length of the common cycle and it is presumed that within a cycle exactly one disassembly lot is begun. Furthermore, the number of remanufacturing lots R and manufacturing lots M (both per cycle) have to be defined. In order to simplify the analysis it is presumed additionally that all remanufacturing lots are of equal size. This presumption is valid for all manufacturing and disassembly lots as well. Analyzing this multi-level inventory system an EOQ-like total cost function TC can be derived.

$$TC = \frac{F}{T} + \frac{\lambda T H}{2} \text{ with } \begin{array}{l} F = K_d + RK_r + MK_m \text{ and} \\ H = \alpha h_d + \frac{R-1}{R}\beta^2 h_r + (\frac{\beta^2}{R} + \frac{(1-\beta)^2}{M})h_s \end{array} \quad (1)$$

By calculating the first derivative of the total cost function with respect to T one obtains the following optimality condition that minimizes the total cost per cycle for a given R and M.

$$TC(R, M) = \sqrt{2\lambda F H} \quad (2)$$

As this function is not defined continuously since R and M have to be integer valued an enumerative procedure is recommended in order to find the optimal solution as presented in [1].

3 Extension of the Model

One of the most critical assumptions of the model from Ferretti and Schulz is that it only allows for one single disassembly lot per cycle to be disassembled. Thus, at least one remanufacturing lot and one manufacturing lot have to be set up in every cycle. This restrictive assumption is made because the obtained total cost function can be analyzed easily. However, following this policy can result in significant errors under specific circumstances. To give an example, if the fixed cost for manufacturing K_m is comparatively large it can be beneficial to pool the necessary requirements for new components of more than one disassembly lot. A more general heuristic solution structure must therefore ensure that the number of remanufacturing and manufacturing lots per cycle need not necessarily be integer multiples of the number of disassembly lots in a cycle. Although this general approach promises better results for any given set of parameters no closed form expression can be found for the total cost function of this generalized problem. That is because both the recoverables and the serviceables inventory lose their characteristic that they have to be zero at the moment a disassembly lot is disassembled. As a consequence, the remanufacturables inventory per cycle cannot be expressed as a simple function that depends on the number of remanufacturing lots R, manufacturing lots M, and disassembly lots D (all per cycle).

In order to determine the total remanufacturables inventory one needs to define when the remanufacturing lots are started. The following iterative procedure can be used for a given T, R, M, and D to calculate those times. The assumptions and notation remain as in the preceding chapter only supplemented by IL that denotes the current remanufacturables inventory level and i as a time index.

1. **start** i=0, IL=0
2. update time index i **and** inventory level IL (every T/D periods there is an inflow of $\lambda\alpha T/D$ units)
3. **if** $i < T$
   a) **then if** at time index i remanufacturing lot size $(\lambda\alpha T/R) < IL$
      then remanufacture one lot **and goto** 2.
      else manufacture one lot of $\lambda(1-\alpha)T/M$ **and goto** 2.
   b) **end if**
4. **end if**
5. **end**

At the beginning (step 1) the time index i as well the remanufacturables inventory level are set to zero. The second step updates those decisions when necessary. Thereby, the time index i is updated in that way that it indicates the next point in time when the serviceables inventory level runs out of stock. The remanufacturables inventory level IL has to be updated after an inflow from disassembly which takes place every T/D periods as well as after every outflow due to a remanufacturing batch. In the third step the procedure checks for a specific i whether there are enough remanufacturable components in inventory for setting up a remanufacturing lot. If the inventory level is not sufficient a manufacturing batch is set up instead. This loop repeats until the end of the common cycle T is reached. The procedure of preferring remanufacturing to manufacturing appears to be conclusive as the immediate manufacturing option shall only be used if there are not enough remanufacturables on hand.

By applying this iterative procedure it is possible to determine the total inventory level throughout the whole cycle as it determines the timing of every remanufacturing lot. Nevertheless, it has to be mentioned that not all parameter combinations are possible without breaking the assumption of equal lot sizes throughout the cycle. For all possible parameter combinations the total cost function can be defined in the same way as for formula 1 except that the part describing the remanufacturables inventory has to be reformulated. An enumerative procedure can be used as in the preceding chapter to find the optimal solution to the problem. For illustrative purposes, figure 2 confronts both procedures. On the left hand side, the old method from [1] is used that defines the best possible heuristic solution respecting all assumptions for D=1, R=2, and M=1. Contrary, the right hand side illustrates the best possible heuristic solution for the same problem with the improved method (for D=2, R=4, and M=1). From the figures it can be concluded that the fixed costs for manufacturing must be compara-

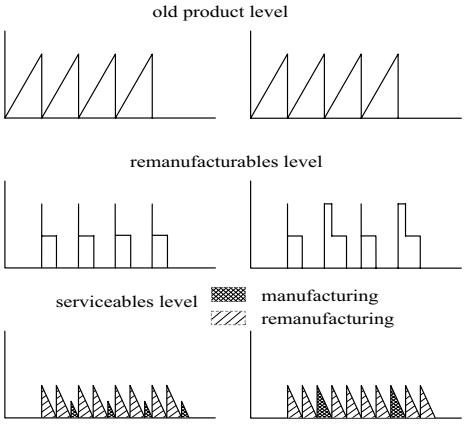


Fig. 2. Comparison of both heuristics

tively high, as it is better to combine the manufacturing requirements of two disassembly lots into one manufacturing batch.

4 Conclusion and Outlook

This contribution introduces and describes a simple production system with remanufacturing and manufacturing options. The following analysis focusses especially on the three-level serial inventory structure of the system and proposed a heuristic for solving this type of problem. Furthermore, this heuristic approach was extended to a more general structure as the solution quality for some parameter constellations appears to be poor. Although losing its simple cost structure some pilot tests have shown that cost improvements of more than 10% in comparison to the old approach are possible.

Future research efforts can concentrate on many different directions. For the sake of brevity only one possible options shall be mentioned here being to compare the introduced heuristics with the optimal solution found by a mixed-integer linear program and with other multi-level lot sizing heuristics.

References

1. Ferretti I, Schulz T (2007) Lot-sizing within a random yield environment. Paper presented at: 8th ISIR Summer School, July 30th - August 3rd 2007. Brescia, Italy.
2. Teunter RH, Bayindir ZP, van den Heuvel W (2006) Dynamic lot sizing for product returns and remanufacturing. International Journal of Production Research, 44(20):4377–4400.

Multicriterial Design of Pharmaceutical Plants in Strategic Plant Management Using Methods of Computational Intelligence

Marion Steven[1] and David Schoebel[2]

[1] Ruhr-Universität Bochum +49 (0) 234 32 23010
marion.steven@ruhr-uni-bochum.de
[2] Bayer Healthcare +49 (0) 214 30 58534
david.schoebel@bayerhealthcare.com

Summary. The **SEMCAIP** (**S**imulation-based **E**volutionary and **M**ultiobjective **C**onfiguration of **A**ctive **I**ngredient **P**lants) decision support system offers support in designing active pharmaceutical ingredient plants within an interdisciplinary group decision-making process. Instead of sequential decision finding, an integrated consensus is reached between different decision makers. The system combines business management models derived from production theory with the process simulation used in process technology. Control of the model calculation with the selection of suitable solutions according to the Pareto principle is exercised by a hierarchical evolution strategy developed in this study. The elaborated decision support system is intended as an instrument to replace conventional, stand-alone use of an investment calculation following the establishment of a technical plant concept.

1 Introduction

It is becoming increasingly difficult in the pharmaceutical industry to find chemical substances that can be developed into new active ingredients for pharmaceutical medications. At the same time, the generic competition has speeded up the pace of its activities. Since the pharmaceutical industry is one of the most research intensive sectors, the necessity to continue investing in research is putting it under increasing pressure to lower costs in other operating areas. The aim was therefore to develop a decision support system for the multicriterial design of active pharmaceutical ingredient plants using methods of computational intelligence as a means of resolving the described problem situation. The **SEMCAIP** (**S**imulation-based **E**volutionary and **M**ultiobjective

Configuration of **A**ctive **I**ngredient **P**lants) decision support system combines business management models derived from production theory with the process simulation used in process technology. The goal is firstly to use the well-proven process technology methods as a basis for determining the technical model variables employed in business management models, and secondly to ensure the acceptance of a solution in a group of decision makers with differing task definitions and specialized backgrounds. Control of the model calculation with the selection of suitable solutions according to the Pareto principle is exercised by a hierarchical evolution strategy developed in this study. Metaheuristics, one of the categories of evolutionary algorithms, were assembled as a modular system from the components known from the literature to solve the specific design problem.

2 Structure of the Decision Support System

The decision support system consists firstly of a technical-economic model to reflect the design of pharmaceutical production plants, and nature-analogous metaheuristics for tuning the decision variables of the model.

The business management partial model is based on a linear activity analysis[3]. The basis of linear activity analysis is the activity. This is understood as a permissible combination of factor input quantities which lead to a certain combination of output quantities when using a given production method. Production processes usually generate not only the intended, marketable products, but also unavoidable accompanying products such as solid, liquid or gaseous residues, waste heat or radiation. Linear activity analysis is an approach that describes production relationships after installation. Since the design task requires a description prior to installation, in this approach the activity analysis is combined with the putty clay model[4]. This production theory model distinguishes between an ex-ante production function which reproduces the choice of technology before installation, and an ex-post production function which reproduces the productivity relationships after the installation of plants. This approach is particularly suitable for solving the problem of designing production plants by integrating the linear

[3] For detailed expositions of activity analysis see e.g.; Debreu (1959), page 46 ff.; Kistner (1993), page 54 ff.; and Steven (1998), page 62 ff.

[4] See Kistner (1993), pp. 133 ff.; Steven (1998), pp. 236 ff.

activity analysis into the putty clay model. The design choice before installation is represented by introducing binary structure parameters.

In this approach, process technology-based process simulation is integrated as a technical partial model to represent the influence of technology-related parameters on the input-output relationships. This process simulation explicitly includes the design of apparatus in the modelling process and thereby satisfies the requirements of initially determining, as a technical partial model, the efficient processes as a basis for the activity analysis. The coefficients for production planning based on the activity analysis can be determined with the aid of process simulation.

To derive the target functions, the quantity level is transformed into the value level. In this approach, economic objectives are represented by the capital value:

$$KW = -A_0 + \sum_{t=1}^{T}(\sum_{j=1}^{m} p_{j,t}x_{j,t} - \sum_{i=1}^{n} q_{i,t}r_{i,t} - \sum_{I=1}^{M} P_{I,t}X_{I,t} + \sum_{J=1}^{N} Q_{J,t}R_{J,t})(1+\gamma)^{-t} \quad (1)$$

The capital value KW is generated by the payments of every single period t. Basis for the calculation are the product prices p_j (j = 1,. . . , m), prices of input factors q_i (I = 1,. . . , n), pollutant releases P_I (I = 1,. . . , M), recycling credits Q_J (J = 1,. . . , N) and the associated volumes x_j, r_i, X_I and R_J. The number of periods is denoted by T, the interest rate by γ.

When formulating ecological target functions it is possible to make use of environmental indicators[5] by placing environmentally relevant data from the input/output balance in relation to each other. In this approach, the following environmental indicators are used as target variables:

[5] See e.g. Pape/Pick/Goebels (2001), page 185.

1. Solvent recovery LM in kg per kg product:

$$LM = \frac{\sum_{t=1}^{T} r_{i,t} - R_{I,t}}{\sum_{t=1}^{T} x_t} \tag{2}$$

2. Raw material R input in kg per kg product:

$$R = \frac{\sum_{t=1}^{T} r_{i,t} - R_{I,t}}{\sum_{t=1}^{T} x_t} \tag{3}$$

3. Emission volume EM in kg per kg product:

$$EM = \frac{\sum_{t=1}^{T} X_{I,t}}{\sum_{t=1}^{T} x_t} \tag{4}$$

A nature-analogous metaheuristic approach for the derivation of solutions is integrated into the technical-economic model. The basis is the evolution strategy, a representative of the evolutionary algorithms which has been extended into a mixed-whole number, multicriterial evolution strategy using approaches known from the literature.

3 Case Example from the Pharmaceutical Industry

The target function values are presented to the decision makers in the representation component of SEMCAIP. For the presentation of the identified alternative designs for active pharmaceutical ingredient plants with reference to the multidimensional objective, all metrically scaled individual target indices KE_i are converted to a corresponding, dimensionless ratio index KV_i for each of the i = 1,. . . , m individual objectives:

$$KV_i = \frac{KE_i^{alt} - KE_i^{worst}}{KE_i^{best} - KE_i^{worst}} \times 100\% \tag{5}$$

The numerator is formed by the difference of the values of the considered and the worst alternatives with reference to a specific individual objective. The denominator represents the difference of the best and the worst alternative. A benchmark oriented approach is therefore being pursued by comparing the identified potential design alternatives for possible active ingredient plants with reference to the individual objectives. The best and worst alternatives are used for the maximum and minimum parameter value (100 % and 0 %). This prevents the scale for the individual objective from becoming too broad and facilitates a visual presentation for the decision makers in the form of a polar diagram:

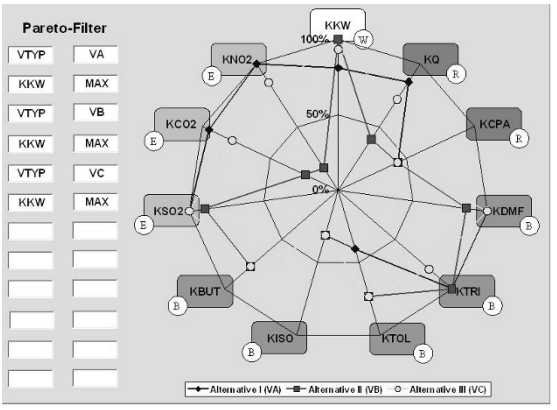


Fig. 1. Target function presentation in Pareto diagram

In the polar diagrams, all index data of an alternative are displayed simultaneously after preliminary selection through a filter. Figure 1 shows three different alternatives with the pertinent index data as an example. The values derive from the Pareto quantity determined by the evolution strategy. They therefore represent non-dominated function values. Further branching is also possible for more detailed analyses.

4 Summary and Prospects

The multicriterial design of active pharmaceutical ingredient plants as a strategic task within plant controlling requires support with suitable instruments. The SEMCAIP decision support system developed

in this study can be used very effectively for the multicriterial design of active pharmaceutical ingredient plants in the early phases of plant development. The great advantage offered by the system developed here is that it provides support for the design of active pharmaceutical ingredient plants within an interdisciplinary group decision-making process. Instead of sequential decision finding, an integrated consensus between different decision makers is reached. The decision support system elaborated in this study therefore has the potential to supersede as an instrument the conventional, stand-alone application of investment calculation following the establishment of a technical plant concept.
As a modular system, it can conceivably be elaborated further in many directions. Firstly, the basic models and objectives can be modified, allowing the system also to be used outside active ingredient production. Under technical model aspects, a particularly valuable addition would be the representation of uncertainty in the model components of the developed decision support system, for example to take into account uncertainties in demand and in process technology based production. Several successful approaches in this area have already been published in the literature[6]. Future studies could examine the extent to which the methods used in these project could be integrated into the SEMCAIP decision support system presented here.

References

1. Debreu, G.: Theory of Value, New York, Wiley-Verlag, 1st edition 1959, 4th edition 1971
2. Kistner, K.-P.: Produktions- und Kostentheorie, Physica-Verlag, Heidelberg, 2nd edition 1993
3. Steven, M.: Produktionstheorie, Gabler-Verlag, Wiesbaden, 1998
4. Pape, J., Pick, E., Goebels, T.: Umweltkennzahlen und -systeme zur Umweltleistungsbewertung, in: Baumast, A., Pape, J. (Ed.), Betriebliches Umweltmanagement, Ulmer-Verlag, Stuttgart, 2001, pp. 178-192
5. Werners, B., Wolf, A.: Simulationsgestützte Steuerung risikobehafteter Projekte, in: Biethan, J. (Ed.): Proceedings zum 9. Symposium: Simulation als betriebliche Entscheidungshilfe: Neuere Werkzeuge und Anwendungen aus der Praxis, Göttingen 2004, pp. 95-114
6. Völkner, P., Werners, B.: A simulation-based decision support system for business process planning, in: Fuzzy Sets and Systems, Vol. 125, No. 3, 2002, pp. 275-287

[6] See e.g. Werners/Wolf (2004) or Völkner/Werners (2002).

Part XVI

Retail, Revenue and Pricing Management

Optimizing Flight and Cruise Occupancy of a Cruise Line

Philipp Kistner, Nadine Rottenbacher, and Klaus Weber

AIDA Cruises, Am Strande 3d, 18055 Rostock, Germany
klaus.weber@aida.de

1 Introduction

The problem described here belongs to the field of revenue managements which origins from the airline industry where it is highly-developed [5, 7]. It has also achieved sophistication to various degrees in other fields of business, e.g. in the cruise industry. We first give a concise introduction to revenue management and its particularities in the cruise industry in section 2 (see [4, 7, 8] for a thorough introduction). Then section 3 introduces and motivates the mathematical model. Results of the optimization are discussed in section 4. Finally, section 5 gives a summary and indicates how this work will be continued. Due to lack of space, this cannot be a comprehensive description. A full description is presented in [3].

2 Revenue Management and Its Particularities in the Cruise Industry

Generally speaking, revenue management aims to optimize the revenue gained from selling a commodity or service which can be characterized as follows [9]:

- Commodity or service capacity is fixed.
- Major part of cost is fixed. Marginal cost is insignificantly low.
- The commodity or service cannot be sold after a specified point in time, i.e., it is "perishable".
- Demand is uncertain.
- The customers' willingness to pay is diverse according to properties of the commodity or service or conditions of their purchase.

A seat on a flight or a cabin of a cruise is a service with above characteristics and, thus, revenue management methods are applicable. Typically following means are applied [9]:

- Market segmentation with respect to willingness to pay and according price discrimination.
- Capacity control by dynamical partition of the amount of commodities or quantity of service at a period of time offered at a specific price.
- Price control by dynamical (re-)setting of price(s).
- Forecast of customer demand according to commodity properties or service conditions or price.

Whereas major airlines maximize their overall network net revenue, cruise lines seek to maximize cruise occupancy – at a "good" price. The main reason for the difference is on-board revenues. On flights they are negligible. Aboard a cruise ship they make a significant part of overall revenue, e.g., for shore excursions, spa treatments, and at bars. As on-board revenues are not well understood, yet, they cannot be taken into account for booking control. Current research aspires to resolve this deficit (e.g., [6]).

In order to carry passengers to remote base ports, AIDA Cruises charters flights. As it bears the full economic risk of the flights, it needs to simultaneously maximize cruise and flight occupancy. It is insufficient to optimize single cruises and their related flights: Due to various passenger journey durations, all outbound and inbound flights and cruises are interconnected.

3 Model Building

The model is introduced step by step. Due to lack of space the model is neither described completely nor in detail, and it is slightly simplified (e.g., group business is omitted). A full description is given in [3].

Figure 1 illustrates a typical situation at AIDA Cruises which is covered by the model: The vessel moves to a base port in the region of destination, e.g., the Caribbean (incoming transit cruise). There it stays for a couple of months and repeats the same route until the end of the season. Then it moves to another region of destination (outgoing transit cruise). Outbound flights carry passengers from various hubs, e.g., in Germany to the base port. Inbound flights bring them back home.

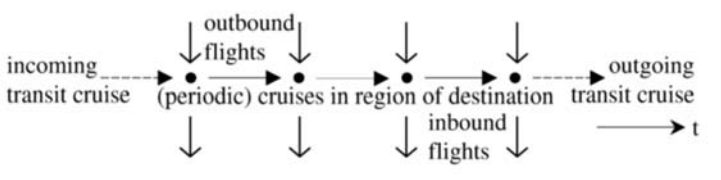


Fig. 1. Situation at the base port of a remote destination

Parameter Sets (Selection)

Let $C \subseteq T_c \times D_c$ the set of cruises starting from the base port. T_c indicates the set of cruise departure dates and D_c the set of cruise durations. $F^+ \subseteq T_f^+ \times H \times FN^+$ and $F^- \subseteq T_f^- \times H \times FN^-$ indicate outbound and inbound flights, respectively, where T_f is the set of flight dates, H is the set of hubs, and FN is the set of flight numbers. (Superscripts +/- indicate outbound/inbound.) $C_{\text{in}}^{\text{trans}}$ and $C_{\text{out}}^{\text{trans}}$ denote the sets of incoming and outgoing transit cruises.

Passengers who book a cruise with flight choose a specific itinerary type:

- "CO" is an itinerary without hotel, i.e. *C*ruise *O*nly.
- "CH" is an itinerary with a hotel stay after the cruise.
- "HC" is an itinerary with a hotel stay before the cruise.
- "NA" designates any itineraries different from the above.

So, $IT := \{\text{CO}, \text{CH}, \text{HC}, \text{NA}\}$ is the set of itinerary types. Possible durations of cruise and hotel stays are given by the itinerary distribution set $D_i^{\text{dist}} \subseteq \mathbb{N}_0^2$. $I := IT \times D_i^{\text{dist}}$ is the itinerary set. Passengers may choose a booking class from set $CL = \{Y, C\}$.

Following sets reflect the interdependency of flights and cruises:

- $FC^+(c, i)$ is the set of outbound flights feeding cruise $c \in C$ with itinerary type $i \in I \backslash \{\text{NA}\}$.
- $FC_{\text{emb}}^+(c, i)$ is the set of outbound flights with passengers embarking for cruise $c \in C$ with itinerary type $i \in I \backslash \{\text{NA}\}$.
- $FF^+(f^-, i)$ is the set of outbound flights feeding inbound flight $f^- \in F^-$ with itinerary type $i \in I \backslash \{\text{NA}\}$ passengers.

Parameters (Selection)

The model comprises two types of parameters. All parameters are related to the date of optimization t_{opt}.

- Supply figures, e.g.,

  - Cruise (only) booking levels: $bkd_c(c)$, $bkd_c^{co}(c)$ for $c \in C$.
  - Empty cruise cabins: $e_{cb}(c)$ for $c \in C$.
  - Current cruise capacity: $cap_c^{\text{curr}}(c)$ for $c \in C$.
  - Outbound flight capacity: $cap_f^+(f^+, cl)$ for $f^+ \in F^+$, $cl \in CL$.
  - Outbound flight booking levels: $bkd_f^+(f^+, cl, i)$ for $f^+ \in F^+$, $cl \in CL$, $i \in I$.
- Demand figures, e.g. cruise demand: $dmd^{emb}(c, cl, i)$ for $c \in C$, $cl \in CL$, $i \in I$.

Optimization Variables (Selection)

The objective of the optimization is twofold:

1. Compute values of booking controls in the reservation system as to maximize occupancy of all cruises and flights. Corresponding optimization variables are:
   - Cruise only contingents: $cont_{\text{co}}(c)$ for $c \in C$.
   - Outbound and inbound flight contingents: $m^+(f^+, cl, i)$, $m^-(f^-, cl, i)$ for $f^+ \in F^+$, $f^- \in F^-$, $cl \in CL$, $i \in I$.
2. Indicate whether preset flights (F^+, F^-) are sufficient, or flights should be added or deleted. Related optimization variables are flight multipliers: $mult_f^+(f^+)$, $mult_f^-(f^-)$ for $f^+ \in F^+$, $f^- \in F^-$.

In addition, model variables exist, e.g., empty berths: $e_b(c)$, empty flight seats: $e_s^+(f^+, cl)$, $e_s^-(f^-, cl)$ and transit cruise contingents: $trans^{in}(c, cl)$, $trans^{out}(c, cl)$ for $c \in C$, $f^+ \in F^+$, $f^- \in F^-$ and $cl \in CL$. All optimization variables are integer and non-negative.

Objective function

It is assumed that the cost of empty cruise berths equals the cost of empty flight seats. Thus, we can reduce the multi-objective problem to a single-objective one: Minimize the sum of empty berths on all cruises and of empty seats on all outbound and inbound flights.

$$\sum_{c \in C} e_b(c) + \sum_{cl \in CL} \sum_{f^+ \in F^+} e_s^+\left(f^+, cl\right) + \sum_{cl \in CL} \sum_{f^- \in F^-} e_s^-\left(f^-, cl\right) \rightarrow \min$$

Constraints (Selection)

Following types of constraints restrict the search space:

- Constraints coupling outbound and inbound flight contingents, e.g.

$$\sum_{f^- \in F^-: f_1^- = t_f^-, f_2^- = h} m^-\left(f^-, cl, i\right) = \sum_{f^+ \in FF^+\left(t_f^-, h, i\right)} m^+\left(f^+, cl, i\right)$$

- Cruise capacity constraints, e.g.

$$cap_c^{curr}(c) \geqslant cont_{\text{co}}(c) + \sum_{cl \in CL} \sum_{i \in I: i_1 \neq \text{NA}} \sum_{f^+ \in FC^+(c,i)} m^+\left(f^+, cl, i\right) + \sum_{cl \in CL} trans^{in}(c, cl) + \sum_{cl \in CL} trans^{out}(c, cl) + e_b(c)$$

- Flights capacity constraints, e.g.

$$e_s^+\left(f^+, cl\right) = mult_f^+\left(f^+\right) \cdot cap_f^+\left(f^+, cl\right) - \sum_{i \in I} m^+\left(f^+, cl, i\right) \quad (1)$$

- Flight booking constraints, e.g.

$$m^+\left(f^+, cl, i\right) \geqslant bkd_f^+\left(f^+, cl, i\right) \quad (2)$$

- Demand constraints, e.g.

$$\sum_{f^+ \in FC^+_{\text{emb}}(c,i)} m^+\left(f^+, cl, i\right) \leqslant \max\left[dmd^{\text{emb}}(c, cl, i), \sum_{f^+ \in FC^+_{\text{emb}}(c,i)} bkd_f^+\left(f^+, cl, i\right)\right]$$

(Index sets are omitted due to lack of space.)

4 Optimization Results

The MIP was implemented in the modeling language Zimpl [2] and solved with ILOG CPLEX [1]. The initial flights in the sets F^+ and F^- are defined by the Flight Operations Department based on experience. At first, the problem was solved for this case, i.e. the flight multipliers $mult_f^+(f^+)$ and $mult_f^-(f^-)$ were fixed to 1. In the next step, constraints were modified to allow the solver to add or delete not more than n outbound and inbound flights, where $1 \leq n \leq 3$. Actually, values beyond 3 are unrealistic.

Solutions indicated both addition of flights and keeping the flights as they are. No solution recommended deletion of flights. Since usually all flights are partially booked this follows from (1), (2) and $e_s^+ \geq 0$.

The economic effect of the optimization is twofold. First, the number of empty seats and berths is minimized. As the more berths are occupied, the more money is spent on-board, this results in higher revenue. The second effect is avoidance of unnecessary additional flights (which saves hundreds of thousands euros). If the solution indicates that flights should be added, then it needs to be evaluated whether the additional passengers aboard compensate for the additional flight cost.

5 Conclusion

For its remote destinations AIDA Cruises needs to manage both occupancy of cruises and charter flights. Means to steer occupancy are booking controls for flight contingents. In the article we desribed an optimization model (MIP) which maximizes occupancy by minimizing the sum of empty berths aboard the cruises and empty seats on flights simultaneously. Additionally, the model indicates whether the foreseen flights are sufficient to fill the ships or additional flights should be ordered. It also indicates when flights shall be deleted. Prototype optimization runs achieved applicable results. In future research we will investigate how the consideration of occupancy could be replaced by the consideration of cost and revenue figures. Furthermore, the possibility to model deletion of flights and re-booking to other flights will be examined.

References

1. ILOG (2007) http://www.ilog.com/products/cplex/
2. Koch T (2004) Rapid Mathematical Programming. ZIB-Report 04-58, Technische Universität Berlin
3. Kistner P, Rottenbacher N, Weber K (2007) Combined Cruise-Flight Capacity Control at a Cruise Line. Unpublished
4. McGill J, van Ryzin G (1999) Revenue Management: Research Overview and Prospects. Transportation Science 33:233–256
5. Pak K, Piersma N (2002) Airline Revenue Management: An Overview of OR Techniques 1982–2001. Econometric Institute Report EI 2002-03, Erasmus University Rotterdam
6. Tromp E (2007) Analysis of Passenger On-Board Revenues of a Cruise Line by means of Data Mining. MA Thesis, Universität Rostock
7. Talluri KT, van Ryzin GJ (2005) The Theory and Practice of Revenue Management. Springer, New York
8. Weatherford LR, Bodily SE (1992) A Taxonomy and Research Overview of Perishable-Asset Revenue Management: Yield Management, Overbooking, and Pricing. Operations Research 40:831–844
9. Weber K (2006) Revenue Management in the Airline and in the Cruise Industry - A Comparison. Presentation at OR 2006, 6–8 September 2006, Karlsruhe

Capacity Investment and Pricing Decisions in a Single-Period, Two-Product-Problem

Sandra Transchel[1], Stefan Minner[1], and David F. Pyke[2]

[1] Mannheim Business School, University of Mannheim, Germany
{sandra.transchel,minner}@bwl.uni-mannheim.de
[2] Tuck School of Business at Dartmouth, Hanover, NH, USA
david.pyke@dartmouth.edu

Summary. We consider a profit-maximizing firm that produces two products with a single capacity. The products are characterized by different demand patterns and the initial capacity is allocated such that the entire demand of one product is satisfied before the other. We develop and analyze a model that determines the optimal initial capacity investment and the selling prices for both products simultaneously. We provide analytical results and develop an algorithm. In a numerical example, we compare this centralized planning approach with decentralized planning where two product managers plan the selling prices and the required production capacity separately but manufacturing produces both products on a common resource. We show that through coordination of pricing and capacity decisions a better capacity utilization can be achieved.

1 Introduction

Over the years, researchers and managers have recognized that the coordination of manufacturing and marketing decisions can improve company performance significantly. However, in most companies marketing and manufacturing operate organizationally separated. Hausman, Montgomery and Roth [2] present an exploratory investigation of the manufacturing-marketing interface. They provide empirical evidence that the ability of marketing and manufacturing to work together matters significantly to business outcomes. Shapiro [5] identifies eight general areas of conflicts that describe the functional interdependencies between manufacturing and marketing.

There is an extensive stream of literature that analyzes simultaneous optimization of price and capacity/inventory in a single-product

case, e.g., see Petruzzi and Dada [3] and Raz and Porteus [4]. This paper studies a capacity investment and pricing problem of a firm that produces two products on a single resource. The products are sold on independent but price-sensitive markets. The products are characterized by a different priority in the production sequence and different demand patterns. One product is an innovative product with an uncertain and moderately price-sensitive demand. The other product is a functional product with a certain but highly price-sensitive demand. This follows Fisher [1] who categorizes products into innovative and functional products. Pricing and capacity decisions are made under uncertainty. After demand realization the demand for innovative products is satisfied before the functional product demand is satisfied.

2 Model

The innovative product is denoted by H (high class) and the functional product is denoted by L (low class). The H product demand is modeled by an additive random, downward-sloping, and linear demand function $D_H := D(P_H, \Psi) = \Psi - bP_H$. Ψ defines a random market potential that is uniformly distributed on the interval $[A, B]$ and $b \geq 0$ defines the slope of the demand function. We assume that for all P_H in a compact decision space $Pr(D_H \leq 0) = 0$. Let $f(z)$ and $F(z)$ be the probability density function and the cumulative density function of Ψ, $\delta = B - A$, and the expected value is denoted by $\mu = \frac{(A+B)}{2}$. The L product is modeled by a deterministic, downward-sloping, and linear price response function with $d_L := d(P_L) = \alpha - \beta P_L$ for $0 \leq P_L \leq \frac{\alpha}{\beta}$ and 0 otherwise. Furthermore, let C define the capacity. We assume that L is only used to fill leftover capacity after satisfying the H demand. Therefore, we constrain the capacity by $C \leq B - bP_H$. The capacity required to produce a unit of product H and L is one capacity unit for both products. The capacity cost is c per unit and the cost for production is c_H per unit of H and c_L per unit of L. For given demands, the profit can be expressed as the difference between revenue, variable manufacturing, and capacity investment costs by the following function:

$$\begin{aligned} \Pi(C, P_H, P_L) = (P_H - c_H) \min\{D_H, C\} \\ + (P_L - c_L) \min\{d_L, \max\{0, C - D_H\}\} - cC. \end{aligned}$$

Taking the expectation value and by substituting $K := C + bP_H$ (see Petruzzi and Dada [3]), $D_H := \Psi - bP_H$, and $d_L := \alpha - \beta P_L$, some algebraic transformations yield the expected profit:

$$\Pi(K, P_H, P_L) = (P_H - c - c_H)(\mu - bP_H) + (P_L - c - c_L)d_L$$
$$- c \int_A^{K-d_L} (K - z - d_L) f(z)dz - (P_L - c - c_L) \int_{K-d_L}^{K} (d_L + z - K) f(z)dz$$
$$- (P_H - c - c_H) \int_K^B (z - K) f(z)dz - (P_L - c - c_L) \int_K^B d_L f(z)dz.$$

Using the uniform distribution and regarding the capacity constraint, we get the following constrained optimization problem

$$\Pi(K, P_H, P_L) = (P_H - c - c_H)(\mu - bP_H) + (P_L - c - c_L)d_L$$
$$- \frac{c}{2\delta}(K - A - \alpha + \beta P_L)^2 - \frac{(P_L - c - c_L)}{2\delta}(\alpha - \beta P_L)^2$$
$$- \frac{(P_L - c - c_L)}{\delta}(\alpha - \beta P_L)(B - K) - \frac{(P_H - c - c_H)}{2\delta}(B - K)^2 \quad (1)$$

$$s.t. \;\; K \leq B. \quad (2)$$

For solving this problem, the *Lagrangian Multiplier* method and the *Karush-Kuhn-Tucker* (KKT) conditions are used. The *Lagrangian function* is $L(K, P_H, P_L, \lambda) = \Pi(K, P_H, P_L) + \lambda(B - K)$ where λ represents the *Lagrangian Multiplier*. The KKT conditions give

$$P_H(K) = \frac{1}{2b}\left(\mu + b(c + c_H) - \frac{(B - K)^2}{\delta}\right), \quad (3)$$

$$P_L(K) = \frac{2}{3\beta}(\alpha + \frac{\beta c_L}{2} - K + A)$$
$$+ \frac{1}{3\beta}\sqrt{4(K - A)^2 + (\alpha - \beta c_L)(\alpha - \beta c_L - 2(K - A))}, \quad (4)$$

$$K(P_H, P_L) = \frac{(P_H - c_H)B + (P_L - c_L)(\alpha - \beta P_L) - (c - \lambda)(B - A)}{(P_H - c_H)}, \quad (5)$$

as well as $\frac{\partial L}{\partial \lambda} \geq 0$, $\lambda \frac{\partial L}{\partial \lambda} = 0$. The derivation of $P_H(K)$, $P_L(K)$, and $K(P_H, P_L)$ is illustrated in Transchel, Minner and Pyke [6].

Proposition 1 *$P_L(K)$ is decreasing and convex in K and $P_L(A) = \frac{\alpha}{\beta}$.*

Proposition 2 *Given P_H and relaxing (2), $K(P_H, P_L)$ is concave in P_L with*

$$K(P_H, \frac{\alpha}{\beta}) = \frac{(P_H - c - c_H)B + cA}{(P_H - c_H)} \in [A, B] \text{ and}$$

$$K(P_H, 0) = \frac{(P_H - c - c_H)B - c_L\alpha + cA}{(P_H - c_H)} \leq K(P_H, \frac{\alpha}{\beta}).$$

The proofs of Proposition 1 and 2 follow directly from (4) and (5). Because K is defined on $[A, B]$, an upper bound P_H^{ub} and a lower bound P_H^{lb} can be derived for P_H with $P_H^{ub} = \frac{1}{2b}(\mu + b(c + c_H))$ and $P_H^{lb} = \max\{(c + c_H), \frac{1}{2b}(\mu + b(c + c_H) - \delta)\}$. Propositions 1 and 2 yield that given P_H, an evaluation of the KKT conditions gives that either there exists an inner solution which solves (4) and (5) (see Figure 1), or the optimal solution is a boundary solution with $K^* = B$, $P_H^* = P_H^{ub}$ and $P_L^* = P_L(B)$ (see Figure 2).

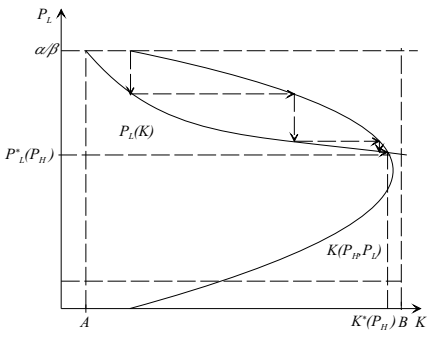


Fig. 1. $P_L(K)$ and $K(P_H, P_L)$ for a given P_H - inner solution

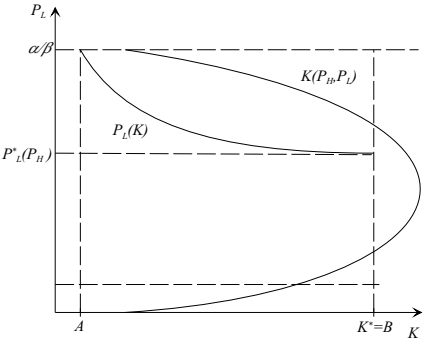


Fig. 2. $P_L(K)$ and $K(P_H, P_L)$ for a given P_H - boundary solution

To solve this problem, the following algorithm can be used. In an outer loop, we enumerate over all P_H from P_H^{lb} to P_H^{ub}. The inner loop uses the results of the above propositions where for each P_H, the optimal $K_{P_H}^*$ and $P_{L_{P_H}}^*$ are determined. Starting from $P_L = \frac{\alpha}{\beta}$, for each P_H a procedure calculates iteratively $K_{P_H}(P_L)$ and $P_{L_{P_H}}(K)$. This procedure converges to $K_{P_H}^*$ and $P_{L_{P_H}}^*$, which is either an inner solution or a boundary solution. Let ϵ be the precision which represents the stopping criterion of the inner loop. In Figure 1 the inner loop is illustrated graphically.

Algorithm

FOR P_H from P_H^{lb} to P_H^{ub} **DO**
  Set $pl_0 := \frac{\alpha}{\beta}$ and $pl_1 := 0$,
  WHILE $|pl_0 - pl_1| > \epsilon$ **DO**
    Calculate $k_1 := K(P_H, pl_0)$ using (5),
    IF $k_1 > B$, **THEN** $K^*_{P_H} = B$ and $P^*_{L_{P_H}} = P_L(B)$,
    ELSE Calculate $pl_1(k_1)$ using (4) **END IF**
    $pl_0 := pl_1$,
  END WHILE
  $K^*_{P_H} = k_1$ and $P^*_{L_{P_H}} = pl_1$
  Calculate $\Pi(K^*_{P_H}, P_H, P^*_{L_{P_H}})$
END FOR

Thus, the optimal profit is $\Pi^* = \max_{P_H}\{\Pi(K^*_{P_H}, P_H, P^*_{L_{P_H}})\}$ as well as $K^* = \text{argmax}\{\Pi^*\}$, $P^*_H = \text{argmax}\{\Pi^*\}$, and $P^*_L = \text{argmax}\{\Pi^*\}$.

3 Numerical Example

In this section, we compare two planning approaches. In a centralized approach as described above, the selling prices and the required capacity of both products are planned simultaneously. In the decentralized case, there exist two product managers who plan the selling prices and the required capacity of each product separately. Due to both products are produced on a common resource, manufacturing installs a capacity which is the sum of the capacity levels planned by both product managers. The index d defines the results of the decentralized decision making. Let $\Psi \sim Unif[250, 750]$, $b = 20$, $\alpha = 500$, $\beta = 40$, $c_H = 3$, and $c_L = 1$. Figures 3 and 4 show the impact of capacity costs c on P^*_H

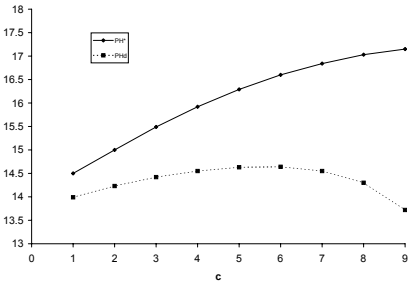


Fig. 3. Impact of c on P^*_H and P^d_H

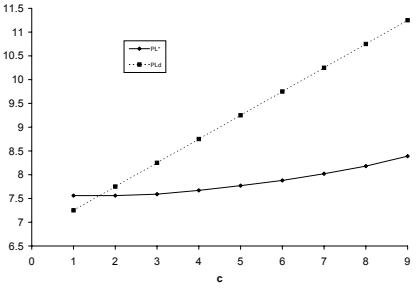


Fig. 4. Impact of c on P^*_L and P^d_L

and P^d_H as well as P^*_L and P^d_L. For H, P^*_H is always larger than P^d_H. For L however, if c is low, P^*_L is larger than P^d_L. In this case a boundary solution is optimal where $K^* = B$ and a decreasing c influences only

the selling prices. If c increases, P_L^* falls below P_L^d. Both pricing effects result from the fact that in the centralized case the decision maker takes into account that leftover capacity from H can be used to produce L. Therefore, P_H^* can be set larger, which actually leads to a lower H demand but each unit of H becomes more beneficial. Because L is used to utilize leftover capacity, the capacity cost c can be regarded as sunk for L. Therefore, P_L^* is only slightly increasing with c. Thus, a centralized planning of prices and capacity leads to a higher flexibility and higher profits than in the decentralized case, where the product managers do not take into account the utilization of leftover capacities.

4 Conclusion

This paper analyzed the problem of a centralized determination of optimal initial capacity investment and the selling prices of two products which are characterized by different demand patterns and a priority in the production sequence. We presented a model that determines the optimal capacity investment and pricing decisions and developed an algorithm. By a numerical example we indicated the impact of capacity costs on the selling prices. Future research will provide a detailed sensitivity analysis regarding the impact of production costs, demand uncertainty, and price-sensitivity. The assumptions of a constrained capacity, additive demand function, and uniformly distributed demand are very restrictive and need to be generalized.

References

1. M. Fisher. What is the right supply chain for your product? *Harvard Business Review*, 75(2):105–116, 1997.
2. W. Hausman, D. Montgomery, and A. Roth. Why should marketing and manufacturing work together? *Journal of Operations Management*, 20(3):241–257, 2002.
3. N. Petruzzi and M. Dada. Pricing and the newsvendor problem: A review with extensions. *Operations Research*, 47(2):183–194, 1999.
4. G. Raz and E. Porteus. A fractile perspective to the joint price/quantity newsvendor model. *Management Science*, 52(11):1764–1777, 2006.
5. B. Shapiro. Can marketing and manufacturing coexist? *Harvard Business Review*, 55(5):104–114, 1977.
6. S. Transchel, S. Minner, and D. Pyke. Coordination of capacity investment and pricing decisions in a two-product-problem with production priorities. Working paper no. 4/2007, University of Mannheim, 2007.

Part XVII

Scheduling and Project Management

Relational Construction of Specific Timetables

Rudolf Berghammer and Britta Kehden

Institut für Informatik, Universität Kiel, Olshausenstraße 40, 24098 Kiel, Germany. {rub | bk}@informatik.uni-kiel.de

1 Introduction

We use relation algebra to model an abstract timetabling problem and to compute solutions. The problem was posed to us by the administration of our university and stems from the current change from the classical German university education system to the undergraduate-graduate system. In particular with regard to the undergraduate education of secondary school teachers this change causes some difficulties. A very serious one is to enable a three years duration of study without to abolish Germany's tradition of (at least) two different fields of study. Exactly this demand is the background of our specific university timetabling problem. We will show how to transform an informal problem description into a formal relation-algebraic model. Using it as starting point, we then develop an algorithm for obtaining solutions. In essence the algorithm is given by a relation-algebraic expression that immediately can be translated into the programming language of the RelView tool. Because of the moderate size of the timetabling problem and the very efficient BDD-implementation of relations in RelView, this even allows to compute all existing solutions of the problem or to message that no solution exists. Due to space restrictions we omit many details. This material will be included in [3].

2 Relation-Algebraic Preliminaries

We denote the set of all relations with domain X and range Y by $[X \leftrightarrow Y]$ and write $R : X \leftrightarrow Y$ instead of $R \in [X \leftrightarrow Y]$. As operations, test, and specific constants we use R^{T} (transposition), $\overline{R}$ (complement), $R \cup S$ (join), $R \cap S$ (meet), RS (composition), $R \subseteq S$ (inclusion),

O (empty relation), L (universal relation) and I (identity relation). We also apply often matrix notation and terminology. Especially, we write $R_{x,y}$ instead of $(x,y) \in R$.

There are some relational possibilities to model sets. First, we will use *vectors*, which are relations v with a singleton set $\mathbf{1} = \{\bot\}$ as range. In these cases we omit the subscript $\bot$. A vector $v : X \leftrightarrow \mathbf{1}$ can be considered as a Boolean column vector and *represents* the subset $\{x \in X \mid v_x\}$ of X. A nonempty vector v is a *point* if $vv^\mathsf{T} \subseteq \mathsf{I}$. This means that it represents an element of its domain. In the matrix model a point $v : X \leftrightarrow \mathbf{1}$ is a Boolean column vector in which exactly one entry is 1. We also will apply *membership-relations* $\mathsf{M} : X \leftrightarrow 2^X$, which are for all $x \in X$ and $Y \in 2^X$ defined by $\mathsf{M}_{x,Y}$ iff $x \in Y$. Third, we will use injection-relations. If Y is a subset of X and $\imath : Y \leftrightarrow X$ such that $\imath_{y,x}$ iff $y = x$ for all $y \in Y$ and $x \in X$, then the vector $\imath^\mathsf{T}\mathsf{L} : X \leftrightarrow \mathbf{1}$ represents Y as a subset of X in the sense above. Clearly, the transition in the other direction is also possible, i.e., the generation of an *injection-relation* $inj(v) : Y \leftrightarrow X$ from the vector representation $v : X \leftrightarrow \mathbf{1}$ of the subset Y of X such that for all $y \in Y$ and $x \in X$ we have $inj(v)_{y,x}$ iff $y = x$. A combination of injection-relations with membership-relations allows a *column-wise representation* of sets of subsets. More specifically, if $v : 2^X \leftrightarrow \mathbf{1}$ represents a subset $\mathcal{S}$ of 2^X in the sense above, then for all $x \in X$ and $Y \in \mathcal{S}$ we get that $(\mathsf{M}\, inj(v)^\mathsf{T})_{x,Y}$ iff $x \in Y$. This means that the elements of $\mathcal{S}$ are represented precisely by the columns of $\mathsf{M}\, inj(v)^\mathsf{T} : X \leftrightarrow \mathcal{S}$.

Given $X \times Y$, there are two projections which decompose $u = \langle u_1, u_2 \rangle$ into its first component u_1 and second component u_2. For our approach it is useful to consider the corresponding *projection relations* $\pi : X \times Y \leftrightarrow X$ and $\rho : X \times Y \leftrightarrow Y$ such that $\pi_{u,x}$ iff $u_1 = x$ and $\rho_{u,y}$ iff $u_2 = y$. Projection relations algebraically allow to specify the *parallel composition* $R \parallel S : X \times X' \leftrightarrow Y \times Y'$ of $R : X \leftrightarrow Y$ and $S : X' \leftrightarrow Y'$ such that $(R \parallel S)_{u,v}$ is equivalent to R_{u_1,v_1} and S_{u_2,v_2}. We get this property if we define $R \parallel S = \pi R \sigma^\mathsf{T} \cap \rho S \tau^\mathsf{T}$, with $\pi : X \times X' \leftrightarrow X$ and $\rho : X \times X' \leftrightarrow X'$ as projection relations on $X \times X'$ and $\sigma : Y \times Y' \leftrightarrow Y$ and $\tau : Y \times Y' \leftrightarrow Y'$ as projection relations on $Y \times Y'$.

We end this section with functions which establish an isomorphism between the Boolean lattices $[X \leftrightarrow Y]$ and $[X \times Y \leftrightarrow \mathbf{1}]$. The direction from $[X \leftrightarrow Y]$ to $[X \times Y \leftrightarrow \mathbf{1}]$ is given by $\mathsf{v}(R) = (\pi R \cap \rho)\mathsf{L}$, and that from $[X \times Y \leftrightarrow \mathbf{1}]$ to $[X \leftrightarrow Y]$ by $\mathsf{r}(v) = \pi^\mathsf{T}(\rho \cap v\mathsf{L}^\mathsf{T})$. Here $\pi : X \times Y \leftrightarrow X$ and $\rho : X \times Y \leftrightarrow Y$ are projection relations and L has type $[Y \leftrightarrow \mathbf{1}]$. Using components the definitions say that $R_{x,y}$ iff $\mathsf{v}(R)_{\langle x,y \rangle}$ and $v_{\langle x,y \rangle}$ iff $\mathsf{r}(v)_{x,y}$.

3 Informal Problem Description

The background is as follows: Presently at our university there exist 34 subjects for the undergraduate education of secondary school teachers. According to the examination regulations each student has to select two subjects. Experience has shown that all possible combinations (pairs of subjects) can be divided into two categories, viz. the very frequently ones (first category) and those which are hardly ever selected (second category). Besides the division of the combinations a further feature of our timetabling problem is the arrangement of the timeslots. There are four disjoint base timeslots $t_1, \ldots, t_4$ of the same duration. But there are some subjects that require two base timeslots. Hence there are two further timeslots t_5 and t_6, where t_5 consists of the hours of t_1 and t_2 and t_6 consists of those of t_3 and t_4. This leads to time conflicts.

Given the input of the timetabling problem in form of the set of subjects, the set of timeslots, the categories of combinations as relations in each case, the availability of the timeslots for the subjects and the time conflict relationship between the timeslots, the goal is to construct a timetable that enables a three years duration of study for the important combinations of the first category and leads to a longer duration of study only in the case of non-important combinations. Concretely this means that we have to construct a function from the subjects to the timeslots such that for all subjects s and timeslots t if s is mapped to t then t is available for s, and there are no conflicts between the courses of two different subjects s and s' if the pair (s, s') constitutes a combination of the first category.

4 Relation-Algebraic Model and Algorithmic Solution

To formalize and generalize the informal problem description to an abstract mathematical timetabling problem, we assume $\mathfrak{S}$ as set of subjects and $\mathfrak{T}$ as set of timeslots. For modeling the partitioning of the pairs of subjects into the two categories, we assume a relation $F : \mathfrak{S} \leftrightarrow \mathfrak{S}$ such that $F_{s,s'}$ iff s and s' form a combination of the first category for all subjects $s, s' \in \mathfrak{S}$. Then F is symmetric ($F = F^{\mathsf{T}}$) and irreflexive ($F \subseteq \overline{\mathsf{I}}$), where the latter property follows from the fact that combinations have to consist of two different subjects. It should be remarked that the relation F suffices for completely describing the two categories introduced in Sect. 3, since the symmetric and irreflexive relation $\overline{F} \cap \overline{\mathsf{I}}$ exactly specifies the pairs which are hardly ever selected. Besides F we

assume a further relation $A : \mathfrak{S} \leftrightarrow \mathfrak{T}$ that specifies availability, i.e., define it component-wise by $A_{s,t}$ iff s can take place in t for all subjects $s \in \mathfrak{S}$ and timeslots $t \in \mathfrak{T}$. And, finally, we assume a third relation $C : \mathfrak{T} \leftrightarrow \mathfrak{T}$ such that $C_{t,t'}$ iff t and t' are in time conflict for all timeslots $t, t' \in \mathfrak{T}$. Since the latter means that the timeslots t and t' contain common hours, C is a reflexive ($\mathsf{I} \subseteq C$) and symmetric relation. The relations $F : \mathfrak{S} \leftrightarrow \mathfrak{S}$, $A : \mathfrak{S} \leftrightarrow \mathfrak{T}$, and $C : \mathfrak{T} \leftrightarrow \mathfrak{T}$ constitute the *input of the university timetabling problem*. Having fixed the input, now we relation-algebraically specify its output.

Definition 1. *Given the three input relations $F : \mathfrak{S} \leftrightarrow \mathfrak{S}$, $A : \mathfrak{S} \leftrightarrow \mathfrak{T}$, and $C : \mathfrak{T} \leftrightarrow \mathfrak{T}$, a relation $S : \mathfrak{S} \leftrightarrow \mathfrak{T}$ is a solution of the university timetabling problem, if $S \subseteq A$, $FSC \subseteq \overline{S}$, $S^{\mathsf{T}}S \subseteq \mathsf{I}$ and $\mathsf{L} \subseteq S\mathsf{L}$.*

These inclusions are a relation-algebraic formalization of the requirements of Sect. 3. Based on an idea of [2], the non-algorithmic specification of a solution of our problem now will be reformulated in such a way that instead of $S : \mathfrak{S} \leftrightarrow \mathfrak{T}$ its so-called *corresponding vector* $\mathsf{v}(S) : \mathfrak{S} \times \mathfrak{T} \leftrightarrow \mathbf{1}$ is used. As we will see later, this change of representation will lead to an algorithmic specification. The following theorem is the key of the approach. It presents a relation-algebraic expression of type $[\mathbf{1} \leftrightarrow \mathbf{1}]$ that depends on a vector $v : \mathfrak{S} \times \mathfrak{T} \leftrightarrow \mathbf{1}$ and evaluates to the universal relation of $[\mathbf{1} \leftrightarrow \mathbf{1}]$ iff v represents a solution S of our problem with input relations F, A and C.

Theorem 1. *Assume F, A and C as in Definition 1, a relation $S : \mathfrak{S} \leftrightarrow \mathfrak{T}$ and a vector $v : \mathfrak{S} \times \mathfrak{T} \leftrightarrow \mathbf{1}$ such that $v = \mathsf{v}(S)$. Then S is a solution of the university timetabling problem iff*

$$\overline{\mathsf{L}((v \cap \overline{\mathsf{v}(A)}) \cup ((F \parallel C)v \cap v) \cup ((\mathsf{I} \parallel \overline{\mathsf{I}})v \cap v) \cup \mathsf{L}\,\overline{\pi^{\mathsf{T}} v})} = \mathsf{L}.$$

Here $\pi : \mathfrak{S} \times \mathfrak{T} \leftrightarrow \mathfrak{T}$ is the first projection relation of $\mathfrak{S} \times \mathfrak{T}$.

Generalizing v to a variable, the left-hand side of the equation of Theorem 1 leads to the following function Φ on relations, where the first L in the definition of Φ has type $[\mathbf{1} \leftrightarrow \mathfrak{S} \times \mathfrak{T}]$ (i.e., is a universal row vector), the second L has domain $\mathbf{1}$ and the same range as X (i.e., is also a universal row vector), the third L has type $[\mathfrak{S} \times \mathfrak{T} \leftrightarrow \mathfrak{S}]$ and X is the name of the variable.

$$\Phi(X) = \overline{\mathsf{L}((X \cap \overline{\mathsf{v}(A)}\,\mathsf{L}) \cup ((F \parallel C)X \cap X) \cup ((\mathsf{I} \parallel \overline{\mathsf{I}})X \cap X) \cup \mathsf{L}\,\overline{\pi^{\mathsf{T}} X})}$$

When applied to $v : \mathfrak{S} \times \mathfrak{T} \leftrightarrow \mathbf{1}$, this function returns $\mathsf{L} : \mathbf{1} \leftrightarrow \mathbf{1}$ iff v corresponds to a solution of our timetabling problem and $\mathsf{O} : \mathbf{1} \leftrightarrow \mathbf{1}$

otherwise. A specific feature is that Φ is defined using the variable X, constant relations, complements, joins, meets and left-compositions only. Hence, it is a *vector predicate* in the sense of [2]. Using matrix terminology this means that if Φ is applied to a relation $R : \mathfrak{S} \times \mathfrak{T} \leftrightarrow Y$ with the family $(v^{(y)} : \mathfrak{S} \times \mathfrak{T} \leftrightarrow \mathbf{1})_{y \in Y}$ of vectors as its columns, then for all $y \in Y$ the y-entry of the row vector $\Phi(R) : \mathbf{1} \leftrightarrow Y$ equals the only entry of $\Phi(v^{(y)})$, i.e., is the truth value 1 if $v^{(y)}$ corresponds to a solution of our problem and the truth value 0 otherwise.

With the specific choice of the powerset $2^{\mathfrak{S} \times \mathfrak{T}}$ for Y and the membership-relation $\mathsf{M} : \mathfrak{S} \times \mathfrak{T} \leftrightarrow 2^{\mathfrak{S} \times \mathfrak{T}}$ for R, we therefore apply Φ in parallel to all possible vectors of type $\mathfrak{S} \times \mathfrak{T} \leftrightarrow \mathbf{1}$, i.e., test all subsets (relations) of $\mathfrak{S} \times \mathfrak{T}$ for being a solution. As a consequence, transposition yields a vector $t = \Phi(\mathsf{M})^{\mathsf{T}} : 2^{\mathfrak{S} \times \mathfrak{T}} \leftrightarrow \mathbf{1}$ such that for all subsets Q of $\mathfrak{S} \times \mathfrak{T}$ the entry t_Q is 1 iff the Q-column of M (considered as a vector $v^{(Q)} : \mathfrak{S} \times \mathfrak{T} \leftrightarrow \mathbf{1}$) corresponds to a solution of our timetabling problem. From t a column-wise representation of all vectors which correspond to a solution of our timetabling problem may be obtained using the relations $inj(t)$ and M in combination with the technique described in Sect. 2. But the vector t also allows to compute a (or even all) single solution(s) in the sense of Definition 1. The procedure is rather simple: First, a point $p \subseteq t$ is selected. Because of the above property, the vector $\mathsf{M}p : \mathfrak{S} \times \mathfrak{T} \leftrightarrow \mathbf{1}$ corresponds to a solution S of our timetabling problem. Now, $\mathsf{r}(\mathsf{M}p) = \mathsf{r}(\mathsf{v}(S)) = S$ shows that S is obtained as $\mathsf{r}(\mathsf{M}p) : \mathfrak{S} \leftrightarrow \mathfrak{T}$.

5 Implementation and Results

At the University of Kiel we have developed a tool for the visualization and manipulation of relations and for relation-algebraic programming, called RelView. It is written in C, uses BDDs for implementing relations, and makes full use of the X-windows graphical user interface. Details and applications can be found, e.g., in [1].

It is straightforward to translate the function Φ into a RelView-program. We have applied the latter to the original problem with 34 subjects and 6 timeslots. Since this led to the membership-relation of size 204×2^{204}, we have not been able to to obtain a result within an adequate time. But the following fact helped to reduce the problem size. There was only one subject (chemistry, abbreviated as c) that required two timeslots. Hence, the model with the six timeslots wasn't appropriate in this case. Instead, chemistry was splitted into two subjects c_1, c_2, so that each of them had to be mapped to one of the four base timeslots. This led to a modified input F', A', C' for the timetable problem.

The relation F' had type $[\mathfrak{S}' \leftrightarrow \mathfrak{S}']$, where $\mathfrak{S}' = (\mathfrak{S} \setminus \{c\}) \cup \{c_1, c_2\}$. For all $s, s' \in \mathfrak{S} \setminus \{c\}$ we defined $F'_{s,s'}$ iff $F_{s,s'}$ and $F'_{c_i,s'}$ iff $F_{c,s'}$ resp. $F'_{s'c_i}$ iff $F_{s',c}$. To guarantee, that c_1 and c_2 are assigned to different timeslots, we finally defined (c_1, c_2) as a combination of the first category, which meant F'_{c_1,c_2} and F'_{c_2,c_1}. The relation $A' : \mathfrak{S}' \leftrightarrow \mathfrak{T}'$, where $\mathfrak{T}' = \{t_1, \ldots, t_4\}$, could be defined as universal relation because now every subject could take place in every timeslot. Since there are no conflicts between base timeslots, finally $C' : \mathfrak{T}' \leftrightarrow \mathfrak{T}'$ could be the identity relation. By modifying the input relations in this way, the function Φ could be used to compute all solutions, since the size of the membership-relation has reduced to 140×2^{140}.

As result of the first computation the solution vector t turned out to be empty, i.e., the given problem was not solvable. With the help of an additional RelView-program we then determined all maximum cliques of F' since large cliques caused the impossibility to find solutions. Step by step 1-entries of F' had been changed to 0, until we obtained an input that led to a nonempty solution vector. The knowledge of the cliques was important for this process to modify the relation F' in a goal-oriented way. We started with 133 combinations in the first category and reduced them to 119 until being successful. This version of F' led to 32 solutions of the timetabling problem. The one that was chosen, enables to study 408 of 561 possible combinations without any overlapping. It also should be mentioned that we could use another property of the given problem to reduce the problem size even more, viz. that each of the four Romanic languages Spanish, Portuguese, Italian and French must be combinable with the other three. This led to a membership-relation of size 124×2^{124}, which allowed to compute all solutions within a few seconds.

References

1. Berghammer R., Neumann F.: RelView– An OBDD-based Computer Algebra system for relations. In: Computer Algebra in Scientific Computing, LNCS 3718, Springer, 40-51 (2005).
2. Kehden B., Evaluating sets of search points using relational algebra, In: Relational Methods in Computer Science, LNCS 4136, Springer, 266-280 (2006)
3. Kehden B., Vectors and vector predicates and their use in the development of relational algorithms (in German). Ph.D. thesis, Univ. of Kiel (to appear 2008)

Alternative IP Models for Sport Leagues Scheduling

Dirk Briskorn

Lehrstuhl für Produktion & Logistik, Christian-Albrechts-Universität zu Kiel, Germany. briskorn@bwl.uni-kiel.de

1 Introduction

Round robin tournaments (RRT) cover a huge variety of real world sports tournaments. Given a set of teams T we restrict all what follows to single RRTs, i.e. each pair of teams $i \in T$ and $j \in T$, $j < i$, meets exactly once and each team $i \in T$ plays exactly once in each period of the tournament. We denote the set of periods by P where $|P| = |T| - 1$. Team $i \in T$ is said to have a break in period $p \in P$ if and only if i plays at home or away, respectively, in $p-1$ and p. In most professional sports leagues in Europe the number of breaks has to be minimized. It is well known that the number of breaks cannot be less than $n - 2$. Moreover, this number can be reached for each even $|T|$. We consider cost $c_{i,j,p}$, $i, j \in T$, $i \neq j$, $p \in P$, for each match of team i at home against team j in period p. The objective is to minimize the overall cost. Models for sports league scheduling have been the topic of extensive research. For the sake of shortness we refuse to give a survey and refer to Briskorn and Drexl [1] for integer programming (IP) models for sports scheduling and to Knust [3] for an extended overview of literature. In section 2 we formulate IP models whose linear programming (LP) relaxation are strengthend in the following by means of valid inequalities. Section 3 provides computational results obtained by employing state of the art solver Cplex and a short conlcusion.

2 Models

2.1 One Class of Break Variables

We propose the following IP model to represent the problem to find the minimum cost single RRT with the minimum number of breaks.

$$\min \sum_{i \in T} \sum_{j \in T \setminus \{i\}} \sum_{p \in P} c_{i,j,p} x_{i,j,p} \tag{1}$$

$$\text{s.t.} \sum_{p \in P} (x_{i,j,p} + x_{j,i,p}) = 1 \qquad \forall\, i,j \in T, i > j \tag{2}$$

$$\sum_{j \in T \setminus \{i\}} (x_{i,j,p} + x_{j,i,p}) = 1 \qquad \forall\, i \in T, p \in P \tag{3}$$

$$\sum_{j \in T \setminus \{i\}} (x_{i,j,p-1} + x_{i,j,p}) - br_{i,p} \leq 1 \qquad \forall\, i \in T, p \in P^{\geq 2} \tag{4}$$

$$\sum_{j \in T \setminus \{i\}} (x_{j,i,p-1} + x_{j,i,p}) - br_{i,p} \leq 1 \qquad \forall\, i \in T, p \in P^{\geq 2} \tag{5}$$

$$\sum_{i \in T} \sum_{p \in P^{\geq 2}} br_{i,p} \leq n - 2 \tag{6}$$

$$x_{i,j,p} \in \{0,1\} \qquad \forall\, i,j \in T, i \neq j, p \in P \tag{7}$$

$$br_{i,p} \in \{0,1\} \qquad \forall\, i \in T, p \in P^{\geq 2} \tag{8}$$

Binary variable $x_{i,j,p}$ is equal to 1 if and only if team i plays at home against team j in period p. Binary variable $br_{i,p}$ equals 1 if and only if team i has a break in period p. Then, the objective function (1) represents the goal of cost minimization. Restrictions (2) and (3) arrange a single RRT. Restrictions (4), (5), and (6) assure that no more than $n-2$ breaks are arranged. As can be seen in Briskorn and Drexl [1] solving this problem using state-of-the-art solver Cplex leads to long runtimes due to the weak lower bound given by the model's LP relaxation. There are lots of implicit restrictions on the venues of a team's matches imposed by the structure of the minimum number of breaks. These structural restrictions are severely weakened in the LP relaxation. Therefore, we propose valid inequalities assuring several of the structural properties of solutions to the IP. By doing this we aim at strengthening the lower bound and, consequently, reducing runtimes. It is well known from Miyashiro et al. [4] that no team can have more than one break and there can not be more than 2 breaks per period if the overall number of breaks is minimum. However, according to the LP relaxation neither of both holds. Therefore, we add restrictions (9) and (10).

$$\sum_{i \in T} br_{i,p} \leq 2 \quad \forall p \in P^{\geq 2} \tag{9}$$

$$\sum_{p \in P^{\geq 2}} br_{i,p} \leq 1 \quad \forall i \in T \tag{10}$$

As shown in Briskorn and Drexl [2] there can not be more than two periods with breaks in a row. Again, this is possible in solutions to the LP relaxation. Consequently, we add restriction (11).

$$\sum_{i \in T} (br_{i,p-2} + br_{i,p-1} + br_{i,p}) \leq 4 \quad \forall p \in P, 4 \leq p \leq n-1 \tag{11}$$

According to restriction (11) no more than four breaks might occur in three consecutive periods. This would mean three periods having breaks in a row. Note that constraint (11) does not prevent three periods to contain a single break each which of course is not possible either. According to Miyashiro et al. [4] the two teams having no break in a single RRT with the minimum number of breaks can be seen as having breaks in the first period. Hence, the first period can be seen as a special case in (11). In order to cover this special case we employ restrictions (12) to (14).

$$\sum_{i \in T} (br_{i,n-2} + br_{i,n-1}) \leq 2 \tag{12}$$

$$\sum_{i \in T} (br_{i,n-1} + br_{i,2}) \leq 2 \tag{13}$$

$$\sum_{i \in T} (br_{i,2} + br_{i,3}) \leq 2 \tag{14}$$

Constraints (12) to (14) prevent two periods (completed to a sequence of three consecutive periods by the first period) from having more than 2 breaks. Note that (12) and (14) imply (11) for $p = n-1$ and $p = 4$. Therefore, we restrict (11) to $5 \leq p \leq n-2$.

2.2 Two Classes of Break Variables

We replace binary variable $br_{i,p}$ by two binary variables $hbr_{i,p}$ and $abr_{i,p}$ for each $i \in T$ and $p \in P^{\geq 2}$. Variable $hbr_{i,p}$ ($abr_{i,p}$) equals 1 if and only if team i has a break at home (away) in period p. Then, restrictions (4) to (6) are replaced by restrictions (15) and (16).

$$\sum_{j \in T \setminus \{i\}} (x_{i,j,p-1} + x_{i,j,p}) - hbr_{i,p} + abr_{i,p} = 1 \quad \forall\, i \in T, p \in P^{\geq 2} \quad (15)$$

$$\sum_{i \in T} \sum_{p \in P^{\geq 2}} (hbr_{i,p} + abr_{i,p}) - (n-2) \qquad = 0 \quad (16)$$

By doubling the number of break variables we obtain several advantages according to the lower bound given by the LP relaxation. Variables $hbr_{i,p}$ and $abr_{i,p}$ equal the number of breaks of i in p at home and away, respectively, while variable $br_{i,p}$ represents an upper bound of the number of breaks of i in p. Therefore, formulating (16) as equation restricts the set of valid constellations of match variables. An equivalent reformulation of (6) is not possible since $br_{i,p} = 1$ does not imply $\sum_{j \in T \setminus \{i\}} (x_{i,j,p-1} + x_{i,j,p}) \neq 1$ for LP solutions. Note that $\sum_{i \in T} hbr_{i,p} = \sum_{i \in T} abr_{i,p}$ holds for each $p \in P$ according to the IP as well as the LP relaxation. Taking this into account, we formulate restrictions (9) and (10) as (17) and (18), here.

$$\sum_{i \in T} hbr_{i,p} \leq 1 \quad \forall p \in P^{\geq 2} \quad (17)$$

$$\sum_{p \in P^{\geq 2}} (hbr_{i,p} + abr_{i,p}) \leq 1 \quad \forall i \in T \quad (18)$$

When reformulating (11) to (14) we, again, take advantage of the more meaningful variables $hbr_{i,p}$ and $abr_{i,p}$. Restriction (19) is tighter than (11) since it prevents three consecutive periods from having a single break each. Constraints (20) to (22) are tighter than (12) to (14), analogously.

$$\sum_{i \in T} (hbr_{i,p-2} + hbr_{i,p-1} + hbr_{i,p}) \leq 2 \quad \forall p \in P, 5 \leq p \leq n-2 \quad (19)$$

$$\sum_{i \in T} (hbr_{i,n-2} + hbr_{i,n-1}) \leq 1 \quad (20)$$

$$\sum_{i \in T} (hbr_{i,n-1} + hbr_{i,2}) \leq 1 \quad (21)$$

$$\sum_{i \in T} (hbr_{i,2} + hbr_{i,3}) \leq 1 \quad (22)$$

3 Computational Results

We executed our test runs on a 3.8 GHz Pentium IV PC with 3 GB RAM. Instances were solved using Cplex 10.0. While instances having

less than 8 teams can be solved to optimality within fractions of seconds instances having 10 teams can not be solved due to lack of memory. Therefore, we focus on instances having 8 teams and solved 30 instances for each model. Moreover, for the most promising models we started test runs for 10 teams but restricted run times to 1 hour. Costs $c_{i,j,p}$ are randomly chosen from $[-10, 10]$. In Table 1 results according to the

Table 1. Run times for one class of break variables

constraints	run times (8)	solved (10)	rel gap (10)
(4),(5),(6)	24.39 sec	60 %	131.3
(4),(5),(6),(9)	20.16 sec	—	—
(4),(5),(6),(10)	13.18 sec	—	—
(4),(5),(6),(9),(10)	14.56 sec	—	—
(4),(5),(6),(11)	20.53 sec	—	—
(4),(5),(6),(11),(12),(13),(14)	18.93 sec	—	—
(4),(5),(6),(11),(12),(13),(14),(10)	17.07 sec	—	—
(4),(5),(6),(11),(12),(13),(14),(9),(10)	15.49 sec	—	—

models with one class of break variables are given. It can be clearly seen that each constellation of valid inequalities leads to lower run times for 8 teams. Restricting the number of breaks to no more than 1 per team turns out to be more effective than restricting the number of breaks per period to be no more than 2. Preventing three periods having breaks in a row is considered in the second section and reduces run times, as well. Moreover, the fraction of runs giving at least one feasible solution for 10 teams and the average relative gap between lower bound and upper bound after 1 hour of run rime is given for the basic model. In Table 2 results according to the models with two classes of break variables are given. The first section shows results where constraint (16) is formulated as inequality. The second section considers (16) as an equality, and we observe that the second variant leads to shorter run times. Considering rows of 3 periods shortens run times. Surprisingly, added to (17) and (18) it does not achieve a run time reduction. As we can see, for each problem at least one feasible solution for 10 teams is given by each of the tested models. Additionally, the relative gap is significantly lower than for the basic model in table 1. Summarizing, models with two classes of break variables and employing both, (17) and (18), lead to best results.

We can conclude that in this paper several models are proposed to find the minimum cost single RRT having the minimum number of

Table 2. Run times for two classes of break variables

constraints	runtimes (8)	solved (10)	rel gap (10)
(15),(16)$^{\leq}$	20.12 sec	—	—
(15),(16)$^{\leq}$,(17)	17.22 sec	—	—
(15),(16)$^{\leq}$,(18)	10.45 sec	100%	41.7
(15),(16)$^{\leq}$,(17),(18)	9.53 sec	100%	36.3
(15),(16)$^{=}$	18.18 sec	—	—
(15),(16)$^{=}$,(17)	17.28 sec	—	—
(15),(16)$^{=}$,(18)	9.65 sec	—	—
(15),(16)$^{=}$,(17),(18)	9.29 sec	100%	34.3
(15),(16)$^{=}$,(19)	20.81 sec	—	—
(15),(16)$^{=}$,(19),(20),(21),(22)	15.27 sec	—	—
(15),(16)$^{=}$,(19),(20),(21),(22),(17),(18)	10.66 sec	100%	36.0

breaks. Valid inequalities were derived from the structure of such tournaments. Each single valid inequality proofs to be run time reducing. The model variant employing two classes of break variables yields lower run times because these variables are more meaningful and, therefore, several inequalities can be formulated tighter.

References

1. D. Briskorn and A. Drexl. Integer Programming Models for Round Robin Tournaments. *Computers & Operations Research*. Forthcoming.
2. D. Briskorn and A. Drexl. Branching Based on Home–Away–Pattern Sets. In K.-H. Waldmann and U. M. Stocker, editors, *Operations Research Proceedings 2006 - Selected Papers of the Annual International Conference of the German Operations Research Society (GOR), Karlsruhe, September 6th - 8th 2006*, pages 523–528. Springer, Berlin, Germany, 2007.
3. S. Knust. Classification of literature on sports scheduling. `http://www.inf.uos.de/knust/sportlit_class/`. (August 02, 2007).
4. R. Miyashiro, H. Iwasaki, and T. Matsui. Characterizing Feasible Pattern Sets with a Minimum Number of Breaks. In E. Burke and P. de Causmaecker, editors, *Proceedings of the 4th International Conference on the Practice and Theory of Automated Timetabling*, Lecture Notes in Computer Science 2740, pages 78–99. Springer, Berlin, Germany, 2003.

Penalising Patterns in Timetables: Novel Integer Programming Formulations

Edmund K. Burke[1], Jakub Mareček[1], Andrew J. Parkes[1], and Hana Rudová[2]

[1] Automated Scheduling, Optimisation and Planning Group
The University of Nottingham School of Computer Science and IT
Jubilee Campus in Wollaton Road, Nottingham NG8 1BB, UK

[2] Masaryk University Faculty of Informatics
Botanická 68a, Brno 602 00, The Czech Republic

Many complex timetabling problems have an underpinning bounded graph colouring component, a pattern penalisation component and a number of side constraints. The bounded graph colouring component corresponds to hard constraints such as "students are in at most one place at one time" and "there is a limited number of rooms" [3]. Despite the hardness of graph colouring, it is often easy to generate feasible colourings. However, real-world timetabling systems [5] have to cope with much more challenging requirements, such as "students should not have gaps in their individual daily timetables", which often make the problem over-constrained. The key to tackling this challenge is a suitable formulation of "soft" constraints, which count and minimise penalties incurred by matches of various patterns. Several integer programming formulations are presented and discussed in this paper.

Throughout the paper, the Udine Course Timetabling Problem is used as an illustrative example of timetabling with soft constraints. The problem has been formulated by Schaerf and Di Gaspero [6, 7] at the University of Udine. Its input can be outlined as follows:

- C, T, R, D, P are sets representing courses, teachers, rooms, days, and periods, respectively
- U is a set representing distinct enrolments in courses ("curricula"), with Inc being the mapping from curricula to (possibly overlapping) subsets of C
- F is a set of pairs $\langle c, p\rangle \in C \times P$, representing deprecated periods p of courses c

- HasEC maps courses to numbers of weekly unit-length events
- HasStud maps courses to numbers of enrolled students
- HasMinD maps courses to recommended minimum numbers of days of instruction per week
- Teaches maps teachers to disjunctive sets of elements of C
- HasCap maps rooms to their capacity
- HasP maps days to tuples of corresponding periods in ascending order.

Given this input, the task is to assign events to rooms and periods so that: for each course, HasEC[c] events are timetabled; no two events take place in the same room at the same period; no two events of one course are timetabled at the same period; events of no two courses in a single curriculum are taught at the same time; events of no two courses taught by a single teacher are timetabled at the same period; for all $\langle c, p\rangle \in F$, events of course c are not taught at period p. The objective is to minimise the following weighted sum: the number of students left without a seat, summed across all events, with weight 1; the number of events timetabled for a curriculum outside of a single consecutive block of two or more events per day, summed across all curricula, with weight 2; the number of missing days of instruction, summed across all courses, with weight 5.

In integer programming, most researchers [9, for example] use a natural assignment-type formulation to model graph colouring. A brief survey of six other possible formulations is given in [4]. For timetabling applications, the following clique-based formulation has been proposed recently [4]: array T of binary decision variables is indexed with periods, rooms and courses. $T[p, r, c]$ being set to 1 indicates course c is being taught in room r at period p. Notice that we do not specify which event of course c is taught at which period. The corresponding hard constraints are:

$$\forall c \in C \sum_{p \in P} \sum_{r \in R} \quad T[p, r, c] = \text{HasEC}[c] \tag{1}$$

$$\begin{matrix}\forall p \in P \\ \forall r \in R\end{matrix} \sum_{c \in C} \quad T[p, r, c] \leq 1 \tag{2}$$

$$\begin{matrix}\forall p \in P \\ \forall c \in C\end{matrix} \sum_{r \in R} \quad T[p, r, c] \leq 1 \tag{3}$$

$$\sum_{\substack{\forall p \in P \\ \forall t \in T}} \sum_{r \in R} \sum_{c \in \text{Teaches}[t]} T[p,r,c] \leq 1 \tag{4}$$

$$\sum_{\substack{\forall p \in P \\ \forall u \in U}} \sum_{r \in R} \sum_{c \in \text{Inc}[u]} T[p,r,c] \leq 1 \tag{5}$$

$$\sum_{\forall \langle c,p \rangle \in F} \sum_{r \in R} T[p,r,c] = 0 \tag{6}$$

Soft constraints in timetabling problems vary widely from institution to institution, but most notably penalise patterns in timetables [10]. Their integer programming formulations, although often crucial for the performance of the model, are still largely unexplored. Although instances of up to two hundred events with dozens of distinct enrolments are now being solved to optimum almost routinely [2], larger instances are still approached only via heuristics.

Out of the three soft constraints in the Udine Course Timetabling problem, the minimisation of the number of students left without a seat can be formulated using a single term in the objective function:

$$\sum_{r \in R} \sum_{p \in P} \sum_{\substack{c \in C \\ \text{HasStud}[c] > \\ \text{HasCap}[r]}} T[p,r,c] \; (\text{HasStud}[c] - \text{HasCap}[r]) \; .$$

The second soft constraint, minimising the number of missing days of instruction summed across all courses, can be formulated using two auxiliary arrays of decision variables. The first binary array, CTT, is indexed with courses and days. $\text{CTT}[c,d]$ being set to 1 indicates there are some events of course c held on day d. The other array of integers, Miss, is indexed with courses. The value of $\text{Miss}[c]$ is bounded below by zero and above by the number of days in a week and represents the number of days course c is short of its recommended days of instruction. This enables addition of the following constraints:

$$\sum_{\substack{\forall c \in C \\ \forall d \in D \\ \forall p \in \text{HasP}[d]}} \sum_{r \in R} T[p,r,c] \leq \text{CTT}[c,d] \tag{7}$$

$$\sum_{\substack{\forall c \in C \\ \forall d \in D}} \sum_{r \in R} \sum_{p \in \text{HasP}[d]} T[p,r,c] \geq \text{CTT}[c,d] \tag{8}$$

$$\sum_{\forall c \in C} \sum_{d \in D} \text{CTT}[c,d] \geq \text{HasMinD}[c] - \text{Miss}[c] \tag{9}$$

The term $5 \sum_{c \in C} \text{Miss}[c]$ can then be added to the objective function.

The natural formulation of penalisation of patterns of classes and free periods in daily timetables for curricula goes "feature by feature", where feature is described relatively to a particular position in a daily timetable, for instance as "a class in period one with period two free". This formulation then uses an auxiliary binary array M, indexed with curricula, days and features, where $M[u, d, f]$ being set to 1 indicates feature f is present in the timetable for curriculum u and day d. The number of features to check, |Check|, obviously depends on the number of periods per day. With four periods per day, we have:

$$\forall u \in U, d \in D, \forall \langle p_1, p_2, p_3, p_4 \rangle \in \mathrm{HasP}[d]$$

$$\sum_{c \in \mathrm{Inc}[u]} \sum_{r \in R} (T[p_1, r, c] - T[p_2, r, c]) \leq M[u, d, 1] \tag{10}$$

$$\sum_{c \in \mathrm{Inc}[u]} \sum_{r \in R} (T[p_4, r, c] - T[p_3, r, c]) \leq M[u, d, 2] \tag{11}$$

$$\sum_{c \in \mathrm{Inc}[u]} \sum_{r \in R} (T[p_2, r, c] - T[p_1, r, c] - T[p_3, r, c]) \leq M[u, d, 3] \tag{12}$$

$$\sum_{c \in \mathrm{Inc}[u]} \sum_{r \in R} (T[p_3, r, c] - T[p_2, r, c] - T[p_4, r, c]) \leq M[u, d, 4] \; . \tag{13}$$

The third term in the objective function is then

$$2 \sum_{u \in U} \sum_{d \in D} \sum_{s \in \mathrm{Check}} M[u, d, s] \; . \tag{14}$$

We refer to this formulation as T, for "traditional". Considerable improvements can be gained by applying the concept of the enumeration of patterns. Let us pre-compute set B of $n+2$ tuples $w, x_1, \ldots, x_{p/|D|}, m$, where x_i is 1 if there is instruction in period i of the daily pattern and -1 otherwise, w is the penalty attached to the pattern, and m is the sum of positive values x_i in the patterns decremented by one. In Formulation E, the array M is replaced with an array W indexed with curricula and days, and constraints (10)–(13) are replaced with:

$$\forall \langle w, x_1, \ldots, x_{p/|D|}, m \rangle \in B \; \forall u \in U \; \forall d \in D \; \forall \langle p_1, p_2, p_3, p_4 \rangle \in \mathrm{HasP}[d]$$

$$w \left(\sum_{i=1}^{p/|D|} x_i \sum_{c \in \mathrm{Inc}[u]} \sum_{r \in R} T[p_i, r, c] \right) - wm \leq W[u, d] \; . \tag{15}$$

The corresponding Term 14 in the objective function is then replaced with $2 \sum_{u \in U} \sum_{d \in D} W[u, d]$. In an alternative Formulation ET, both

arrays M and W are used, together with constraints (10)-(15). Term 14 in the objective function is replaced with:

$$\sum_{u \in U} \sum_{d \in D} \sum_{s \in \text{Check}} M[u, d, s] + \sum_{u \in U} \sum_{d \in D} W[u, d] \; . \tag{16}$$

This could perhaps guide the search better than terms involving only M or only W. In yet another Formulation ETP, Formulation ET is strengthened using constraints:

$$\sum_{u \in U} \sum_{d \in D} W[u, d] - \sum_{u \in U} \sum_{d \in D} \sum_{s \in \text{Check}} M[u, d, s] = 0 \tag{17}$$

$$\forall_{u \in U} \sum_{d \in D} W[u, d] - \sum_{d \in D} \sum_{s \in \text{Check}} M[u, d, s] = 0 \tag{18}$$

$$\forall_{u \in U} \forall_{d \in D} W[u, d] - \sum_{s \in \text{Check}} M[u, d, s] = 0 \; . \tag{19}$$

Finally in Formulation TP, the original Formulation T is strengthened using constraints resembling (15), whose right-hand sides are replaced with $\sum_{s \in \text{Check}} M[u, d, s]$.

These five formulations, together with Formulation C of the decision version of graph colouring, have been encoded in Zimpl [8] and tested on four real-life instances from the University of Udine School of Engineering [6] and 6 semi-randomly generated instances. The results in Table 1 have been obtained with SCIP 0.82 using SoPlex [1], running

Table 1. Performance of five formulations of pattern penalisation: instance name, number of events, occupancy in percent, and either the run time needed to reach the optimality, or the gap remaining after two hours of solving. A blank indicates no feasible solution has been found

Instance	Ev.	Occ.	C	T	TP	E	ET	ETP
udine1	207	86	(2 s)	(1417 s)	(1231 s)	500.00 %	(655 s)	(916 s)
udine2	223	93	(10 s)	42/0	49/0		44/0	48/0
udine3	252	97	(25 s)	638.21 %	470.54 %			755.46 %
udine4	250	100	(28 s)	4800.00 %	4500.00 %			
rand1	100	70	(3 s)	(345 s)	(434 s)	37/0	(1134 s)	(567 s)
rand2	100	70	(2 s)	(888 s)	(610 s)	0.28 %	(615 s)	0.02 %
rand3	100	70	(1 s)	(561 s)	(440 s)	0.52 %	(751 s)	(1228 s)
rand4	200	70	(21 s)	1.62 %	2.25 %		2.25 %	0.50 %
rand5	200	70	(30 s)	2.89 %	0.11 %		0.03 %	0.31 %
rand6	200	70	(25 s)	0.57 %	0.82 %			0.79 %

on Linux-based Sun V20z with dual Opteron 248 and 2 GB of memory. Notice that all constraints were given explicitly in these experiments. Results for other semi-randomly generated instances, together with the instances themselves, are available from the authors' website[3].

The formulation of soft constraints penalising patterns in timetables of individual students or groups of students is crucial for the performance of integer programming formulations of timetabling with soft constraints. The presented formulations penalise patterns not only by feature, but also by enumeration of patterns over daily timetables. They might prove to be a good starting point for further research into branch-and-cut algorithms for timetabling with soft constraints.

Acknowledgements. The authors are grateful to Andrea Schaerf and Luca Di Gaspero, who maintain the Udine CTT Problem. Hana Rudová has been, in part, supported by Project MSM0021622419 of MŠMT.

References

1. T. Achterberg. *Constraint Integer Programming.* PhD thesis, Berlin, 2007.
2. P. Avella and I. Vasil'ev. A computational study of a cutting plane algorithm for university course timetabling. *J. Scheduling*, 8(6):497–514, 2005.
3. E. K. Burke, D. de Werra, and J. H. Kingston. Applications to timetabling. In *Handbook of Graph Theory*, pages 445–474. CRC, London, UK, 2004.
4. E. K. Burke, J. Mareček, A. J. Parkes, and H. Rudová. On a clique-based integer programming formulation of vertex colouring with applications in course timetabling. Technical report, 2007. at http://arxiv.org/abs/0710.3603.
5. E. K. Burke and S. Petrovic. Recent research directions in automated timetabling. *European J. Oper. Res.*, 140(2):266–280, 2002.
6. L. D. Gaspero and A. Schaerf. Multi neighborhood local search with application to the course timetabling problem. In *Practice and Theory of Automated Timetabling, PATAT 2002*, pages 262–275, Berlin, 2003. Springer.
7. L. D. Gaspero and A. Schaerf. Neighborhood portfolio approach for local search applied to timetabling problems. *J. Math. Model. Algorithms*, 5(1):65–89, 2006.
8. T. Koch. *Rapid Mathematical Programming.* PhD thesis, Berlin, 2004.
9. I. Méndez-Díaz and P. Zabala. A cutting plane algorithm for graph coloring. *Discrete App. Math.*, 2008. In press.
10. H. Rudová and K. Murray. University course timetabling with soft constraints. In *Practice and Theory of Automated Timetabling, PATAT 2002*, pages 310–328, Berlin, 2003. Springer.

[3] `http://cs.nott.ac.uk/~jxm/timetabling`

Online Optimization of a Color Sorting Assembly Buffer Using Ant Colony Optimization

Stephan A. Hartmann and Thomas A. Runkler

Siemens AG, Corporate Technology
Information and Communications, CT IC 4
Otto-Hahn-Ring 6, 81730 Munich - Germany
stephan.hartmann.ext@siemens.com, thomas.runkler@siemens.com

Summary. In this paper we present an ant based approach for the problem of scheduling a color sorting assembly buffer online. In the automotive industry, all car bodies are painted at a paint shop, where it is important that the number of color changes is minimized. The car bodies on the assembly line are unsorted with respect to their color, thus a color sorting assembly buffer may be used to reduce the number of color changes. The problem of finding an optimal strategy for controlling a *color sorting assembly buffer* (CSAB) consists of two closely related sub-problems: the *color retrieval problem* (CRP) and the *color storage problem* (CSP). Their combination, the *color storage and retrieval problem* (CSRP) is NP-complete, existing methods are not applicable on larger problems. In this paper we introduce two ant colony optimization (ACO) algorithms that probabilistically solve the CRP and the CSP, respectively. They significantly outperform the conventional rule based approach.

1 Introduction

In the automotive industry car bodies are mostly produced in assembly belt production. When using the *build to order* strategy each body gets assigned to a customer order at the beginning of the manufacturing process, hence the color of a car is specified before assembly. However, the bodies on the assembly line are not sorted by desired color.

The car bodies are sequentially processed in the painting station. It is crucial to minimize the number of color changes to reduce costs and time resulting from these changes. We consider the average *color block length* (CBL), i. e. the average number of consecutive bodies with the same color as a appropriate measure for the quality of a car body sequence to be painted. After the body shop the CBL is quite short, so

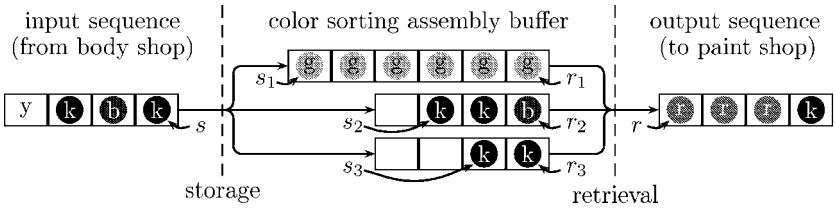


Fig. 1. Scheme of a CSAB and its connection to the production environment. The letters s, r, s_i and r_i are defined in Section 2

the goal is to increase the CBL for the paint shop by using a CSAB as shown in Figure 1. The CSAB is a group of parallel FIFO queues (buffer lines). Each line may contain bodies with different desired colors. There are two control decisions that affect the average CBL: Into which line should an incoming body be stored (*color storage problem*, CSP) and from which line should the next body be retrieved for painting (*color retrieval problem*, CRP). Both problems are highly connected and often treated together as the *color storage and retrieval problem* (CSRP) which is NP-complete [3]. Epping and Hochstätter have suggested solutions for the CSRP and exact dynamic algorithm in [2]. However, the authors "doubt the practical use" since the feasible problem size is extremely small due to the enormous computational costs. The idea of a CSAB was also applied for accelerating graphic processing in [4]. The ACO used here was introduced by Dorigo [1] in 1992 and since then was applied to various discrete optimization problems, such as logistic systems [5]. In ACO a set of agents (artificial ants) probabilistically construct solutions S through of a collective memory, the *pheromones* (stored in matrix τ), together with a problem specific heuristic η.

2 Rule Based Approach for the CSRP

Voß [6] introduced a rule based approach for the CSRP that proved to be superior to other heuristics for this problem. For each body it considers the color and its "age" represented by the cycle number in which the body gets stored in the buffer. We denote the input and output sequence with seq_{in} and seq_{out}, respectively, and define:

s := color of the next body in seq_{in} to be stored.
r := color of the last body in seq_{out} that was retrieved from the buffer.
s_i := color of the body in buffer line i that was stored last.
r_i := color of the body in buffer line i that will be retrieved next.

Retrieval Rules:

a_i := age of the body in buffer line i that was stored last.
b_i := age of the body in buffer line i that will be retrieved next.
n_i := number of bodies with color r_i at the retrieval end of line i.

R1 Choose the storage line l with $r_l = r$ and b_l being minimal.
R2 If R1 is not applicable, choose the storage line l with minimal b_l.

Storage Rules:

S1 Choose a non-full storage line l with $s_l = s$ and a_i being maximal.
S2 If S1 is not applicable, choose empty line.
S3 If S2 is not applicable, try to optimize the overall performance. For each storage line i count the number of bodies with color s_i in seq_{in}. Choose non-full storage line l with the lowest count.

3 Supplementation of Storage Rules

In order to improve the performance we suggest additional storage rules that exploit two circumstances.

First, when R1 is applied it may happen, however, that within the next $n = \sum_{i:r_i=r} n_i$ bodies in the input queue there are some of color r. In this case we suggest to store the next n bodies in a way such that we can append the r-colored bodies to the color block that is retrieved at that moment. This is possible if one line, the *fast line*, is only filled with color r or empty. Then the r-colored bodies "within reach" for retrieval in the input queue pass all earlier assigned non-r-colored bodies in the buffer through this fast line.

Second, assume that in storage line i there are two different colored blocks ($r_i \neq s_i$) and storage line j has the same color as s_i. This implies a temporary order relation in the buffer, i. e. $r_i \prec s_i$. If we now want to store a body with $s = r_i$ we would violate this order when storing it in line j since $s_i = s_j \prec s = r_i$. This would result in at least two separate r_i-colored blocks in seq_{out} after retrieval. Thus when storing a new body in the buffer we select a storage line k without r_i-colored bodies, if possible. By this means we determine all lines that s can be storied in without violating the order relation. Note that this order relation is transitive, i. e. $r_i \prec s_i = r_j, r_j \prec s_j \Rightarrow r_i \prec s_j$. We derive the following rules (applied in this order until one is applicable: S1, S1a, S2, S2a, S3):

S1a If S1 is not applicable, choose single-colored line with color r (the fast line must not be used, if it exists). Note that color $s \neq r$.
S2a If S2 is not applicable, apply S3 on all lines that would not violate the temporary order of the buffer when a body is stored.

4 ACO for CRP

Retrieval rule R1 makes sure that the color block is continued if possible. Retrieval rule R2, however, might produce suboptimal color block lengths depending on the color distribution on the input and storage lines. To increase the CBL we replace R2 by a modified version of a special ACO algorithm called Min-Max-Ant-System (MMAS).

Whenever rule R1 does not apply, for determining the next retrieval color we assume that seq_{out} is empty and seq_{in} is frozen, i. e. no bodies are stored into the buffer. We want to empty the whole buffer optimally. The next color to be retrieved is the first color of the optimal temporary output sequence T that results from emptying the buffer.

In the pheromone matrix τ each position represents the probability of body d being retrieved to the q-th position in T. In each iteration of the algorithm T is constructed as follows: If there are r_i that hold $c \not\prec r_i$ (transitive!) for any color c in the buffer, one of lines i (otherwise one of *all* lines) is chosen by selecting the line of body d with probability $p_{dq} = \frac{\tau_{dq}}{\sum \tau_{dq}}$. Only these r_i can be stored in a row in T. After making a color change in T, try to apply rule R1. Repeat these two steps until the buffer is empty. For each color block cb in T we determine its length l_{cb} and v_{cb}, that is defined as the number of bodies that precede cb in T. T is evaluated by: $ev = \sum_{cb \in T} l_{cb} \cdot v_{cb}$. Less color blocks result in less summands. The products ensure that larger color blocks tend to be retrieved earlier in order not to block the buffer. The algorithm proceeds by choosing out of n temporary output sequences the one with the smallest ev and updates the pheronomes corresponding to this T by $\tau_{dq} = \tau_{dq} + 5$ (see Algorithm 1).

Algorithm 1: ACO-CRP

```
Initialization of the pheromone matrix τ with 1;
while iteration < maximum number of iterations do
    for n ants do
        for d = 1 to (# of bodies in the buffer) do
            Try to apply rule R1;
            Otherwise determine lines i with c ⊀ r_i, ∀c in the buffer;
            Choose one line of i probabilistically using values of r_i in τ;
        Compute evaluation ev = Σ_{cb∈T} l_cb · v_cb;
    Choose out of the last n constructed T the one with smallest ev.
    Update τ: τ_dq = τ_dq + 5 when body d at q-th position in T;
Return best solution T_opt;
```

5 ACO for CSP

We now use rule R1 and the ACO-CRP for retrieval and rules S1–S2a for storage decisions. S3 is replaced by ACO. Assume that the retrieving order T_{opt} of all car bodies in the buffer is estimated by ACO-CRP whenever Algorithm 2 is called. Furthermore assume that seq_{in} will shorten when a body is retrieved, i. e. it is running empty. The algorithm is similar to ACO-CRP. It determines which body from seq_{in} should be stored in which line until all colors from T_{opt} are retrieved from the buffer. The algorithm runs as follows: If rules S1–S2a do not apply, one of the lines i is chosen probabilistically using the pheromone matrix $\hat{\tau}$. If the buffer already contains s, only lines are considered for storage that do not violate the buffer's order relation (as in ACO-CRP), if possible. One body is retrieved according to T_{opt}. This is repeated until all colors from T_{opt} are retrieved in the given order. The assignment A is then evaluated: ev stands for the total number of color blocks in the buffer and seq_{out} and should be minimized. The update is analogous to ACO-CRP except that all pheromones in $\hat{\tau}$ additionally evaporate with the factor 0.9.

Algorithm 2: ACO for CSP

```
Initialization of the pheromone matrix τ̂ with 1;
while iteration < maximum number of iterations do
    for n ants do
        while T_opt is not retrieved completely do
            Try to apply rules S1–S2a;
            Determine lines i which s could be assigned to without
            violating the induced order relation;
            Choose one line of i probabilistically using τ̂;
        Compute evaluation of A: ev = # blocks in buffer and in seq_out;
    Choose out of the last n constructed A the one with smallest ev.
    Update τ̂: τ̂_dq = τ̂_dq + 5 when body d at q-th storage line in A;
    τ̂ = τ̂ · 0.9
Return best solution A_opt;
```

6 Computational Results

The results in this section are based on real world data that was provided by a major German car manufacturer. The data included the number and length of the buffer lines and an input color sequences with length of 47000 bodies covering about two months of production. For ACO-CRP (ACO-CSP) we used 20 (10) ants for 500 (850) iterations. The buffer's filling ratio is optimal around 70 % with respect to

the CBL. In the beginning no car bodies were retrieved until the filling ratio was reached.

Table 1 shows that results from the ACO approaches are 34.9 % and 41.8 % better than the pure rule based approach. Although they need much more computational time this is irrelevant here.

Table 1. Results for proposed combinations of storage and retrieval strategies

Storage	R[1, 2]	R1, ACO-CRP		R1, ACO-CRP	
Retrieval	S[1, 2, 3]	S[1, 1a, 2, 2a, 3]		S[1, 1a, 2, 2a], ACO-CSP	
avg. CBL	9.89	13.34	(+ 34.9 %)	**14.02**	(+41.8 %)
avg. time [sec]	**184**	11451	(+6123 %)	76111	(+41265 %)

7 Conclusions and Future Work

The ACO approaches clearly outperform the rule based approach. Nevertheless, there are circumstances that would result in ACO-CSP being too slow (e. g. a larger buffer). In ACO-CSP the computation of the best storage decision is consuming most of the time, thus future work should focus on accelerating this part, e. g. by making it dynamic, using gathered information about storage decisions for several stages. This can potentially be used in order to find a very good solution much faster.

References

1. M. Dorigo and T. Stützle. *Ant Colony Optimization.* MIT Press, 2004.
2. T. Epping and W. Hochstätter. Sorting with line storage systems. In *Operations Research Proceedings*, pages 235–240, 2002.
3. C. Geiger, R. Hunstock, G. Lehrenfeld, W. Müller, J. Quintanilla, C. Tahedel, and A. Weber. Visual modeling and 3D-representation with a complete visual programming language – a case study in manufacturing. In *IEEE Symposium on Visual Languages*, pages 304–307, 1996.
4. J. Krokowski, H. Räcke, C. Sohler, and M. Westermann. Reducing state changes with pipline buffer. In *International Fall Workshop Vision, Modeling, Visualization*, pages 217–224, 2004.
5. C. A. Silva, J. M. Sousa, T. A. Runkler, and J. Sá da Costa. Distributed optimization in supply–chain management using ant colony optimization. *International Journal of Systems Science*, 37(8):503–512, 2006.
6. S. Voß and S. Spiekermann. Simulation von Farbsortierspeichern in der Automobilindustrie. In *Fachtagung: Simulation und Animation für Planung, Bildung und Präsentation*, 1996.

Scheduling of Tests on Vehicle Prototypes Using Constraint and Integer Programming

Kamol Limtanyakul

Robotics Research Institute, Dortmund University, Otto-Hahn Str. 8, 44227 Dortmund, Germany. kamol.limtanyakul@uni-dortmund.de

Summary. We address a scheduling problem in the automobile industry where hundreds of tests must be performed on vehicle prototypes. The objective is to minimize the number of required prototypes, while all tests are executed with respect to several kinds of constraints. We apply Constraint Programming (CP) to solve the complete problem. Then to complement CP in finding a good lower bound, we separately solve an Integer Programming (IP) model which is simplified to cope with the large scale of the problems involved. This model is based on two main principles: set covering to select prototypes with suitable components for all tests; and energetic reasoning to determine the number of prototypes definitely required over the time intervals between the distinct values of the release and due dates. Computational results show that CP can achieve good feasible solutions as confirmed by the results obtained from solving the simplified IP model.

1 Introduction

A car manufacturer must conduct several hundred tests on vehicle prototypes before starting mass production of a vehicle series. The scheduling problem arises as most prototypes can be used for several tests if the tests are appropriately arranged. The prototypes are handmade and thus very expensive. Therefore, the manufacturer aims at minimizing the number of required prototypes, while all tests are on schedule. Moreover, there are hundreds of prototype variants depending on the combinations of various components. Each test must be assigned to an appropriate prototype variant according to its component requirements. Tests are also subjected to several temporal constraints and some specific constraints, for instance, any crash test must always be the last test executed on a prototype. The details of this problem are discussed in [6].

There are some similar projects related to test scheduling in the automobile industry. The problem characteristics are slightly different but they have the common objective of reducing the cost of prototype production. Due to the large problem size, either a multi-stage mathematical programming method [5] or a heuristic approach [2] are suggested. Integer Programming (IP) can be applicable for small size cases [2]. Constraint Programming (CP) is then used to minimize the makespan as the manufacturer also wants to complete all tests as soon as possible [6]. As a tuning parameter, the number of prototypes may be initially set to a large value and iteratively reduced to find the minimum value which still leads to a valid schedule. We found that CP can achieve a feasible solution even for the largest problem [6]. However, CP can prove that the problem is infeasible only if the given number of prototypes is very low. Between both extremes there is a large unknown gap where the feasibility cannot be proved within reasonable time.

In this paper, we formulate another CP model which directly minimizes the number of required prototypes. Although CP helps us get a feasible solution, we still need to find a good lower bound for the optimality gap measurement. CP is not suitable in this situation as it can tighten the lower bound only for nodes locally explored in the same branch of the search tree. On the contrary, IP provides a global perspective as the branch-and-bound process always raises the lower bound. Hence, we suggest a simplified IP model which can be solved for a large instance and provide a solution regarded as a lower bound for this problem.

2 Formal Problem Description

To describe the complete problem, we define the following notations: let V, I, and J be the sets of prototype variants, prototypes, and tests, with $|V| = l$, $|I| = m$, and $|J| = n$. Each prototype $i \in I$ belongs to variant $v_i \in V$. $M_j \subseteq V$ is the set of prototype variants that can perform test $j \in J$. $N_v \subseteq J$ is the set of tests that variant v can execute.

Each test $j \in J$ has a processing time p_j, a release date r_j, and a due date d_j. Each prototype $i \in I$ has an available time a_i plus a set-up time s_{v_i} depending on variant v_i. Also, the construction of variant v might be delayed due to the delivery time y_v of some components which cannot be compensated for within prototype manufacturing. Therefore, a test cannot be executed on prototype i before time $\max\{a_i + s_{v_i}, y_{v_i}\}$.

Furthermore, we consider the following relations between two different tests $j, k \in J$: $j \prec k$ iff test j must be completed before test k is

started; $j \sim k$ iff tests j and k must be executed on the same prototype; and $j \asymp k$ iff tests j and k cannot be executed on the same prototype. $J_{Last} \subseteq J$ is the set of tests that must not be followed by any other test on the same prototype. Finally, the objective is to determine the smallest number of prototypes m and their corresponding variants such that no late test occurs.

3 Complete CP Model

It is common in CP to define a resource constraint by using the term cumulative($\mathbf{t}, \mathbf{p}, \mathbf{c}, C$), where $\mathbf{t} = [t_1, .., t_n]$, $\mathbf{p} = [p_1, .., p_n]$, $\mathbf{c} = [c_1, .., c_n]$ are the vectors of starting time, processing time, consumption rate of jobs, and C is the machine capacity. The constraint is satisfied if the condition $\sum_{j \in J_t} c_j \leq C$ holds for all time instances t in the valid time frame, where $J_t = \{j \in J | t_j \leq t < t_j + p_j\}$ is the set of tasks that are in process at time t. Therefore, the total consumption of all jobs $j \in J_t$ does not exceed the capacity C. In our case, each prototype has capacity $C = 1$ and each test requires a single prototype, $c_j = 1, \forall j \in J$.

Moreover, let us define the following variables. First, the starting time t_j of a test j must be in the interval $[r_j, ..., d_j - p_j]$ to obey the release date and due date constraints. Second, $x_j \in I, \forall j \in J$ represents the allocation of test j to prototype i. Third, $v_i \in G \cup \{0\}, \forall i \in I$ indicates the variant of prototype i. Here, we introduce a dummy variant 0 to help us represent the objective function. If prototype i is given variant 0 ($v_i = 0$), it cannot be used to execute any test because variant $\{0\} \notin M_j, \forall j \in J$. Also, this prototype becomes redundant as it does not perform any test. Finally, the binary variable $z_i, \forall i \in I$ indicates whether prototype i is required or not. $z_i = 1$ if this prototype is still necessary for any test and it does not belong to the dummy variant, i.e. $v_i \neq 0$. As the objective function is to minimize the total number of required prototypes, it means the dummy variant should be assigned to as many prototypes as possible in the given set I. The complete CP model is to

$$\text{minimize } \sum_{i \in I} z_i$$

subject to

$$v_i \neq 0 \Rightarrow \ z_i = 1 \qquad \forall i \in I \qquad (1)$$

$$\text{cumulative}(t_j | x_j = i, p_j | x_j = i, c_j = 1 | x_j = i, 1) \ \forall i \in I \qquad (2)$$

$$v_i \notin M_j \Rightarrow x_j \neq i \qquad \forall i \in I, \ \forall j \in J \qquad (3)$$

$$x_j = i \Rightarrow t_j \geq a_i + s_{v_i} \qquad \forall i \in I,\ \forall j \in J \quad (4)$$

$$x_j = i \Rightarrow t_j \geq y_{v_i} \qquad \forall i \in I,\ \forall j \in J \quad (5)$$

$$t_j + p_j \leq t_k \qquad \forall j \prec k,\ j,k \in J \quad (6)$$

$$x_j = x_k \qquad \forall j \sim k,\ j,k \in J \quad (7)$$

$$x_j \neq x_k \qquad \forall j \asymp k,\ j,k \in J \quad (8)$$

$$x_j = x_k \Rightarrow\ t_j \geq t_k + p_k \qquad \forall j \in J_{\text{Last}}, \forall k \in J. \quad (9)$$

Constraint (1) ensures that prototype i is required if its corresponding variant is not the dummy variant. Constraint (2) is a resource constraint with the term $(t_j | x_j = i)$ denoting the tuple of starting times for tests that are assigned to prototype i. Constraint (3) prevents a test from being assigned to a prototype whose variant does not belong to the eligibility set of the test. Constraint (4) and Constraint (5) ensure that a test on machine i cannot start before the availability of the prototype according to our model. Precedence constraints are represented by Constraint (6). Constraint (7) forces that any pair of tests (j, k) with $j \sim k$ is executed on the same machine. Furthermore, tests j and k with $j \asymp k$ must be processed on different machines which is achieved by Constraint (8). Finally, Constraint (9) ensures that test $j \in J_{Last}$ will be the last job executed on the machine to which it is allocated.

4 Simplified IP Model

We develop the simplified model which is based on two principles. First, the selection of a set of prototype variants for all tests is a set covering problem, as mentioned by Lockledge et al. [5]. To tighten the model further, we modify the IP model suggested by Hooker [3]. The original purpose of his model is to find a tight lower bound of the number of late jobs according to the given capacity of cumulative resources. But in our case the capacity of resource (or the number of prototypes) is minimized such that no late job (or test) happens. In fact, this concept is similar to energetic reasoning used for constraint propagation in CP [1].

The main idea here is to consider the minimum processing time of jobs for any interval between times t_1 and t_2. Let $J(t_1, t_2)$ be a set of jobs whose release dates and due dates lie between $t_1 \leq r_j$ and $d_j \leq t_2$. In case of a single machine, the necessary condition to ensure that no late job occurs is $\sum_{j \in J(t_1,t_2)} p_j \leq t_2 - t_1$. Also, the important values for t_1 are in set $\bar{R} = \{\bar{r}_1, ..., \bar{r}_{n_r}\}$ which contains the distinct elements in set of release dates $R = \{r_1, ..., r_n\}$ and for t_2 are in set $\bar{D} = \{\bar{d}_1, ..., \bar{d}_{n_d}\}$ which contains the distinct elements in set of due

dates $D = \{d_1, ..., d_n\}$. As the model does not consider the effect of processing jobs at every point in time, we regard the solution of this IP model as the lower bound of the minimum number of prototypes.

Let us define the following variables: the binary variable w_{vj} equals one if test j is assigned to prototype variant v, otherwise zero; integer variable y_v is the required number of each variant. The IP model is to

minimize $\sum_{v \in V} y_v$

subject to

$$\sum_{v \in M_j} w_{vj} = 1 \qquad \forall j \in J \tag{10}$$

$$y_v \geq w_{vj} \qquad \forall j \in J, \forall v \in V \tag{11}$$

$$\sum_{j \in N_v \cap J(t_1,t_2)} w_{vj} p_j \leq y_v (t_2 - t_1) \quad \forall v \in V, t_1 \in \bar{R}, t_2 \in \bar{D}, t_2 > t_1 \tag{12}$$

$$w_{vj} = w_{vk} \qquad \forall j \sim k, \forall v \in M_j \cap M_k \tag{13}$$

$$w_{vj} + w_{vk} \leq y_v \qquad \forall j \asymp k, \forall v \in M_j \cap M_k \tag{14}$$

$$\sum_{j \in J_{\text{Last}} \cap N_v} w_{vj} \leq y_v \qquad \forall v \in V\ . \tag{15}$$

The model minimizes the total number of required prototypes. Constraint (10) assigns each test to exactly one appropriate variant. Constraint (11) ensures that the number of each variant is greater than one if it is required by at least one test. The energy consumption requirement is represented in Constraint (12). Furthermore, we can loosely include some additional constraints into the model. Constraint (13) ensures that tests executed on the same prototype must also be assigned to the same variant. If tests which must be performed on different prototypes are allocated to the same variant, Constraint (14) demands at least two vehicles for that variant. Similarly, Constraint (15) ensures that the number of last jobs assigned to any variant must be less than the number of that prototype variant.

Moreover, note that it is possible to consider the available time of each prototype as the processing time of m additional jobs at the starting time. However, we have to neglect this part in order to keep the model simple and remain solvable in the case of quite large instances.

5 Computational Results

We apply our approach to solve problems with data obtained from a real-life test scenario. There are four data sets of different sizes ranging

from about 40 to almost 500 tests. We use OPL Studio 3.7 [4] to formulate and solve both CP and IP models. All computations are performed on a Pentium IV, 3.0 GHz. The computation time is limited at 1 hour.

The computational results are shown in Table 1. As CP solves the complete problem, its solution represents the valid number of required prototypes. For small problems like data set 1, the optimal solution can be found. But when a problem grows bigger, the search tree explodes rapidly. Only feasible solutions can then be achieved. In case of solving the simplified IP model, we obtain the lower bound values of the number of required prototypes. The results show that the gaps between solutions obtained from CP and IP models are just one prototype at the most. It means not only CP can already achieve good feasible solutions but also the simplified IP model can provide close lower bounds.

Table 1. Minimizing the number of required prototypes

Data	n	Complete CP		Simplified IP	
		#Prototype	Time(s)	#Prototype	Time(s)
1	41	5[a]	0.98	5[a]	2.74
2	100	5	1.11[b]	4[a]	0.38
3	231	19	14.82[b]	18[a]	7.63
4	486	111	242.03[b]	110[a]	844.21

[a] Optimal solution
[b] Computation time of the achieved solution

References

1. P. Baptiste, C. L. Pape and W. Nuijten (2001) Constraint-Based Scheduling: applying constraint programming to scheduling problems. Kluwer.
2. J.-H. Bartels and J. Zimmermann (2005) Scheduling Tests in Automotive R&D Projects, In: Operations Research Proceedings 2005, Springer, volume 11, pages 661–666.
3. J. Hooker (2005) A Hybrid Method for the Planning and Scheduling. Constraints 10(4):385–401.
4. ILOG S.A. (2003) ILOG OPL Studio 3.7 Language Manual.
5. J. Lockledge, D. Mihailidis, J. Sidelko, and K. Chelst (2002) Prototype fleet optimization model. Journal of the Operational Research Society 53:833–841.
6. K. Limtanyakul, and U. Schwiegelshohn (2007) Scheduling Tests on Vehicle Prototypes using Constraint Programming. In: Proceedings of the 3rd Multidisciplinary International Scheduling Conference: Theory and Applications, pages 336–343.

Complexity of Project Scheduling Problem with Nonrenewable Resources

Vladimir V. Servakh[1] and Tatyana A. Shcherbinina[2]

[1] Omsk Branch of Sobolev Institute of Mathematics SB RAS, Pevtsov street 13, 644099 Omsk, Russia. svv_usa@rambler.ru
[2] Department of Mathematics, Omsk State University, Mira prospect 55A, 644077 Omsk, Russia. shcherbinina@bk.ru

1 Introduction

In the paper we consider the project scheduling problem (PSP) under resource constraints. By the project, we mean some set of jobs the processing of which is aimed at achievement of a definite purpose. Examples of such projects are: mining, development and reconstruction of territories, military and space programs. The planning of the project consists in setting the starting times of job processing. The difficulty is connected with limitation of various material, labor and financial resources.

Project scheduling problem with renewable resources is NP-hard in the strong sense [1]. In paper [2] is proved polynomial solvability of the project scheduling problem with nonrenewable resources and deadline with criterion of minimization of the total completion time of the project. In this work we consider two variants of the project scheduling problem with nonrenewable resources: minimization of average completion time of the jobs of the project and maximization of net present value (NPV). We prove that these problems are strongly NP – hard by reduction the maximum clique problem.

2 Problem Definition

Problem 1

Let the project consist of a set of interrelated jobs $V = \{1, 2, \ldots, N\}$. The interrelation is set by the technology of processing of the project and defined by the relation of a kind $i \to j$ where job j can not start

before completion of the job i. The given structure can be presented in a form of a directed acyclic graph $G = (V, E)$, where V is the set of vertexes, and $E = \{(i, j)|i, j \in V, i \to j\}$ is the set of edges. There are M kinds of nonrenewable resources that may be used to perform a job. At any moment there are R_m units of a resource of a kind m. Each job $j \in V$ is characterized by its processing time $p_j \in Z^+$ and the number of units r_{mj} of the resource of the kind m $(m = 1, \ldots, M)$. Preemptions of jobs are not allowed. It is required to find the starting times of job processing in the project satisfying the technological order and restrictions on resources, minimizing some goal function.

By s_j we denote the starting time of the job $j \in V$. Then $c_j = s_j + p_j$ is the time of its completion. Vector $S = (s_1, s_2, \ldots, s_N)$ is the *schedule* job processing of the project. If all p_j are integers, it is possible to consider only the schedules with integer s_j $(j \in V)$. We denote $N_t = \{j \in V|s_j < t \leq c_j\}$ as a set of jobs performed within an interval of $(t-1, t], t \in Z^+$.

Schedule S is feasible if:

1. The partial order job processing is held:

$$s_i + p_i \leq s_j, \quad (i, j) \in E; \tag{1}$$

2. At any moment the constraints on resources are satisfied:

$$\sum_{t'=1}^{t} \sum_{j \in N_{t'}} r_j(t' - s_j) \leq \sum_{t'=1}^{t} K(t'), \;\; t \in Z^+ \backslash \{0\}. \tag{2}$$

The purpose of the project is the minimization of average completion time project jobs:

$$C_\Sigma = \frac{1}{N} \sum_{j \in V} c_j \to \min. \tag{3}$$

Problem 2

Let the project consist of a set of interrelated jobs $V = \{1, 2, \ldots, N\}$. The interrelation is given by a directed acyclic graph $G = (V, E)$, where V is the set of vertexes, and $E = \{(i, j)|i, j \in V, i \to j\}$ is the set of edges. There is one kind of nonrenewable resources (financial). At any moment t there are $K(t)$ units of a financial resource. Each job $j \in V$ is characterized by its processing time $p_j \in Z^+$ and the cash flow c_j, where $c_j(t)$ is a cash flow $j \in V$ at moment t, $t = 0, 1, \ldots, p_j$. So if $c_j(\tau) < 0$ then at time τ investments exceed the profit; if $c_j(\tau) > 0$

then the profit is greater than investments by that value. Preemptions of jobs are not allowed.

The purpose of the project is *Net Present Value (NPV)* criterion. The NPV for the cash-flow of the job $j \in V$ is determined by the formula:

$$NPV_j = \sum_{t=0}^{p_j} \frac{c_j(t)}{(1+r_0)^t},$$

where r_0 is the market value of capital. If $NPV_j > 0$, that job $j \in V$ should be considered *profitable*.

The objective is to compute a schedule $S = \{s_j\}$ that meets all resource and precedence constraints and criterion of NPV:

$$NPV(S) = \sum_{j \in V} \frac{NPV_j}{(1+r_0)^{s_j}} \to \max, \tag{4}$$

with constraints on the variable of $\{s_j | j \in V, s_j \in Z^+\}$:

1. The partial order job processing is held:

$$s_i + p_i \le s_j, \quad (i,j) \in E; \tag{5}$$

2. At any time moment $t^* \ge 0$ the constraints on resources are satisfied:

$$\sum_{t=0}^{t^*} \sum_{j \in N_t} \frac{c_j(t-s_j)}{(1+r_0)^t} \le \sum_{t=0}^{t^*} \frac{K(t)}{(1+r_0)^t}, \quad t^* \in Z^+. \tag{6}$$

3 NP-hardness of the Project Scheduling Problem

We consider a maximum clique problem. A clique of a graph $\Gamma = (V^\Gamma, E^\Gamma)$ is a subset V^0 of V^Γ, such that every two nodes in V^0 are joined by an edge of E^Γ. The maximum clique problem consists of finding y_0 as the largest cardinality of a clique. We denote $v = |V^\Gamma|$ and $e = |E^\Gamma|$. This problem is NP-hard in the strong sense [3].

Project scheduling problem (1)– (3) is correspondent to the following problem of recognition: if there exists such an feasible schedule S^0, with respect to G which can perform $j \in V$ project jobs so that $C_{\sum}(S^0) \le y$ for the given value of y.

We show that the maximum clique problem is reduced polynomially to the above formulated problem of recognition. Its basic idea is offered [4].

The set V is formed as follows. Each vertex j of the graph Γ is associated with a vertex job V_j, and each edge of the graph Γ is associated with an edge job $E_{i,j}$. Let us define a strict order relation on the constructed set V: $V_i \to E_{i,j}$ and $V_j \to E_{i,j}$ for all $(i,j) \in E^{\Gamma}$. Let each vertex job and each edge one consume one unit of a nonrenewable resource. Let the volume of the resource available be as $K(1) = y_0$ for the first period of time, as $K(2) = (v - y_0) + \frac{y_0(y_0-1)}{2}$ for the second period and as $K(3) = e - \frac{y_0(y_0-1)}{2}$ for the third period.

Therefore while searching feasible schedules with respect to G we can restrict our search to examining such schedules in which:

a) y_0 vertex jobs will be performed during time interval of $[0,1]$;
b) $e - \frac{y_0(y_0-1)}{2}$ edge jobs will be performed during time interval of $[2,3]$;
c) the remaining vertex jobs ($v - y_0$ pieces) and the edge jobs ($\frac{y_0(y_0-1)}{2}$ pieces) will be performed during time interval of $[1,2]$.

Assume $y = y_0 + 2((v - y_0) + \frac{y_0(y_0-1)}{2}) + 3(e - \frac{y_0(y_0-1)}{2})$.

Let us suppose that in the graph Γ there exists a clique containing not less than y_0 vertices. Then, there is such a clique $\Gamma' = (V', E')$, that $|V'| = y_0$. We will consider the schedule S^0, in which:

a) y_0 vertex jobs corresponding to the vertices of Γ' have been performed during the time interval of $[0,1]$;
b) all the rest vertex and edge jobs corresponding to the edges of the graph Γ' (there are $\frac{y_0(y_0-1)}{2}$ pieces of them since the clique exists) are performed in the time interval of $[1,2]$. The jobs are performed in the order that is feasible with respect to G (G is a graph of the relation reduction $\to$);
c) the remaining edge jobs corresponding to the graph $\Gamma \backslash \Gamma'$ are performed during the time interval of $[2,3]$.

Since the resource requirements for edge and vertex jobs have been fulfilled, and all the available resources have been used it means that S^0 is an feasible schedule with respect to G and $C_{\sum}(S^0) = y_0 + 2((v - y_0) + \frac{y_0(y_0-1)}{2}) + 3(e - \frac{y_0(y_0-1)}{2}) = y$.

For the problem (1)–(3) let S^0 be the optimal schedule that satisfies the condition that $C_{\sum}(S^0) \le y_0 + 2((v-y_0) + \frac{y_0(y_0-1)}{2}) + 3(e - \frac{y_0(y_0-1)}{2})$. Then during the time interval of $[0,1]$ exactly y_0 vertex jobs are performed. Let us denote the set of these vertex jobs as V^0. Edge jobs

cannot be performed during this time interval since any edge job, $E_{i,j}$ is preceded with two vertex jobs, V_i and V_j, where $(i,j) \in E$. During the time interval of $[1,2]$ the number of $\frac{y_0(y_0-1)}{2}$ edge jobs are performed. We denote this set as E^0. The remaining vertex jobs of the set $V \backslash V^0$ are performed as well. During the time interval of $[2,3]$ the vertex jobs from the set $V \backslash V^0$ cannot be performed since there are no isolated vertices in the graph G. Hence, the remaining edge jobs are performed during the time interval of $[2,3]$. Since the schedule S^0 is feasible, edge jobs E^0 are not associated with vertex jobs from $V \backslash V^0$. But then edge jobs from E^0 are associated with them. Then it follows that we have a subgraph consisting of vertices corresponding to the set V^0 and edges corresponding to the set E^0. But since $|E^0| = \frac{y_0(y_0-1)}{2}$) and $|V^0| = y_0$, it follows that this subgraph is complete.

Finally, we come to the conclusion that the schedule S^0 satisfying the above-mentioned condition (a)–(c) will be feasible if and only if the graph Γ has a clique possessing not less than y_0 vertices. Under this condition vertex jobs corresponding to the vertices of the clique are to be performed during the time interval of $[0,1]$, and edge jobs corresponding to the edges of the clique are to be performed during the time interval of $[1,2]$. Reduction implementation requires performing not more than $O((v+e)^2)$ operations.

The same result is obtained for problem 2.

References

1. Garey M.R., Johnson D.S. (1975) Complexity result for multiprocessor scheduling under resource constraints. SIAM J. Comput. 4(4): 397-411.
2. Gimadi E.Kh., Zaljubovsky V.V., Sevastianov S.V. (2000) Polynomial solvability of the project scheduling problem with accumulative resources and deadline, (in Russia). Diskret. Analiz i Issled. Oper. 7(1), Ser. 2, 9-34.
3. Karp R.M. (1972) Reducibility among combinatorial problems. In R.E. Miller, J.W. Thatcher (eds.), Complexity of Computer Computations, Plenum Press, New York, 85-103.
4. Brucker P., Lenstra J.K., Rinnooy Kan A.H.G. (1975) Complexity of machine scheduling problems. Math. Cent. Afd. Math. Beslisk. Amsterdam. BW 43.

Part XVIII

Simulation, System Dynamics and Dynamic Modelling

Optimizing in Graphs with Expensive Computation of Edge Weights

Frank Noé[1], Marcus Oswald[2], and Gerhard Reinelt[2]

[1] DFG Research Center "Matheon", FU Berlin, Arnimallee 6, 14195 Berlin, Germany. noe@math.fu-berlin.de
[2] University of Heidelberg, Institute for Computer Science, Im Neuenheimer Feld 368, 69120 Heidelberg, Germany

1 Introduction

Much research has gone into the development of fast graph optimization algorithms and many problems, such as shortest-path [1] or minimum-cut [2], can now be routinely solved for large graphs. In many practical applications, *e.g.* in bioinformatics or computational chemistry [3, 4], the solution of graph optimization problems is very important, but hampered by the fact that the graph is not completely known. Especially edge weights, which may represent, for example, reaction rates in a reactive network, are often unknown or only known within some error bounds. Usually, methods for determining these edge weights are available, but the precise determination of a single edge weight may require long computations or expensive experiments. The objective is thus not to minimize the runtime of the graph optimization problem with a given set of edge weights, but instead to minimize the number of edge weights that need to be determined in order to be able to solve the graph optimization problem.

To our knowledge, this problem has so far not been addressed in operation research literature. In the present paper we present a simple heuristic for solving the graph problem while trying to compute exact weights only for few edges and to avoid determining the weights of other edges which have no or little impact on the problem. The algorithms are applied to shortest paths and minimum cuts in biomolecular reaction networks and the results demonstrate that only few edge weights need to be determined in order to solve these graph optimization problems. This encourages further research in this area.

In the following, let $G = (V, E, c)$ denote a graph with node set V, edge set E and a vector of edge weights c. The weight of an edge $e = uv$ is denoted by c_e or c_{uv}. In a general problem setting we have a set $\mathcal{I} \subseteq 2^E$ of feasible solutions and an objective function $f : 2^E \to \mathbb{R}$. The optimization problem consists of finding $I^* \in I$ such that $f(I^*) = \min\{f(I) \mid I \in \mathcal{I}\}$. We denote by $f^*(G)$ the optimum objective function value and by $\mathrm{OPT}(G)$ an optimal edge set. In this paper we require that the objective function f has the following *edge-monotonous property*: if the weight c_e of one edge $e \in E$ is increased, then f remains the same or increases. If c_e is decreased, then f remains the same or decreases. We start with no exact edge weights being given, but instead some finite lower and upper bounds l_e and u_e are available for every edge e. Correspondingly we can define the two graphs $G_l = (V, E, l)$ and $G_u = (V, E, u)$. Due to the monotonicity property, $f^*(G_l) \le f^*(G) \le f^*(G_u)$. Furthermore, we are given some means for refining bounds by replacing l_e and u_e of a given edge e by new bounds l'_e and u'_e such that $l'_e \ge l_e$ and $u'_e \le u_e$.

2 Algorithms

2.1 Basic Algorithm

From the given lower and upper bounds on the edge weights of $G = (V, E, c)$ we can construct the two graphs $G_l = (V, E, l)$ and $G_u = (V, E, u)$. We assume that we have an algorithm to compute $\mathrm{OPT}(G_l)$ and $\mathrm{OPT}(G_u)$ and a method for refining edge weight bounds as described above. The following iterative algorithm determines $\mathrm{OPT}(G)$.

Algorithm 1 Compute $\mathrm{OPT}(G)$

(1) Compute $\mathrm{OPT}(G_l)$ and $\mathrm{OPT}(G_u)$. Output $f^*(G_u)$ and $f^*(G_l)$.

(2) If $f^*(G_u) - f^*(G_l) \le \Delta$, then `return`$(\mathrm{OPT}(G_l))$.

(3) Select an undetermined edge $h \in \mathrm{OPT}(G_l)$ with maximum weight $l_h = \max\{l_e \mid e \in \mathrm{OPT}(G_l) \text{ and } l_e < u_e\}$. (We call such an edge a *critical edge*). Refine h and `goto` (1).

Theorem 1. *Assuming that at most n_{refine} edge refinements are necessary to exactly determine its weight ($l_e = u_e$), then, for $\Delta = 0$, Algorithm 1 computes OPT(G) in finitely many steps.*

Proof. Edges with determined weight are never selected as critical edges in step (3). Therefore each edge can only be determined once. The algorithm terminates in step (2) at the latest when all edges are determined. (Although it is expected to terminate earlier.) Thus at most $|E| \cdot n_{\text{refine}}$ edge refinements are required.
The correctness of the algorithm follows immediately from the fact that $f^*(G_l) \leq f^*(G) \leq f^*(G_u)$ always holds. Thus the theorem is proved. □

For $\Delta > 0$ the algorithm is no longer exact, but will return an approximate solution I with $f^*(I) \leq f^*(G) + \Delta$. In many practical cases, such an approximate solution may be sufficient and save considerable amounts of CPU time.

2.2 Parallelization

Algorithm 1 can be parallelized in a rather straightforward way. To establish a communication between the individual processes, a "database" of bounds is required which every process has read and write access to. The database contains the vectors L and U of the current lower and upper edge weight bounds, and an edge is marked "busy" when a process is about to change its weight. Every processor keeps own graphs G_l and G_u and executes Algorithm 2.

Theorem 2. *The parallel algorithm computes OPT(G) in finite time.*

Proof. Assume that each edge weight refinement takes finite time. When iterating the loop (2)–(7), one edge weight is refined in each iteration and since each edge is determined only once in a given process, this will terminate after at most $|E| \cdot n_{\text{refine}}$ cycles. Loop (2)–(6) is iterated only if all $e \in \text{OPT}(G_l)$ are determined. If $F = \emptyset$, then the algorithm will terminate in the next iteration in step (5). If $F \neq \emptyset$, then the process will wait in (5) and other processes are currently in step (6), determining the edges $e \in \text{OPT}(G_l)$ which are also in F. As these other process will finish the determination in finite time and afterwards update L and U, eventually $F = \emptyset$ and the process terminates in the next iteration in (5). □

Only in the first part of step (7) possibly incorrect edge weights are assigned and added to F. They are only removed from F in step (2) if their true values have been determined. Thus, $F = \emptyset$ only if all weight bounds L and U are correct, and only in this case the algorithm returns. Thus Algorithm 2 produces the same result as Algorithm 1 and the theorem is proven. □

Algorithm 2 Parallel computation of OPT(G)

(1) Let $F = \emptyset$. (F is a set of edges with estimated weights.)
(2) Remove every member e from F that is not *busy*.
(3) Update $l_e = L_e$ and $u_e = U_e$, for all edges $e \in E$.
(4) Compute OPT(G_l) and OPT(G_u).
(5) If all edge weights of OPT(G_l) are determined, *i.e.*, $l_e = u_e$, for all $e \in$ OPT(G_l), then
    (5.1) If $F = \emptyset$ `return`(OPT(G_l)), otherwise wait for some time interval τ.
(6) Select an undetermined edge h with maximum weight from OPT(G_l). If no such edge exists, `goto` 2.
(7) Distinguish the following cases:
    (7.1) If h is *busy*, then assign a hypothetical edge weight to h and set $F = F \cup \{h\}$.
    (7.2) If h is not *busy*, then mark h as *busy*, refine h, set $U_h := u_h$, $L_h := l_h$, and mark h as not *busy*.
    Then `goto` 2

3 Applications to Molecular Transition Networks

We describe an application of shortest paths in the computation of the dynamics of biomolecules where it is very expensive to obtain exact edge weights [3, 4], typically requiring minutes to hours of CPU time for each edge weight. Biomolecules, such as proteins, undergo transitions between metastable "end-states" which correspond to different atomic coordinates and have different biological functions. For example, `Ras p21` is a cell signaling protein which exists in an *active* state that promotes cell growth and an *inactive* state that inhibits cell growth. These states are "connected" via intermediate states which are typically short-lived and have no particular biological function.

In a *transition network*, a state is modeled as a vertex $v \in V$, and a possible transition between a pair of states is modeled by an (undirected) edge $uv \in E$.

3.1 Shortest Paths

When the mean residence times are used as edge weights in a transition network, the shortest paths between two given vertices u and v represent the most populated transition pathways for the molecule to change between the two associated structures [4].

Using the algorithms presented in Section 2, we have computed the best paths of four different transitions in biomolecules while determining only a small number of edge weights: the pathways for the $\alpha_L \rightleftharpoons \beta$, $\beta \rightleftharpoons \alpha_R$ and $\alpha_L \rightleftharpoons \alpha_R$ transitions in the $\texttt{Ala}_8$ peptide and the active⇌inactive transition in the `Ras p21` molecular switch. Detailed descriptions of these molecular systems can be found in [4, 3]. For computing the shortest paths for a given set of edge weights, Dijkstra's algorithm was employed [1].

We first used trivial initial edge weight bounds (0 and ∞). For the $\texttt{Ala}_8$ and `Ras p21` networks the number of edge weights required to be computes are up to three orders of magnitude below $|E|$. Table 1 displays in the second column the number of edges of the networks and in the third column the actual number of refinement steps of Algorithm 1.

Table 1. Number of determined edge weights for shortest path computation

	$\lvert E\rvert$	n_{ec}, normal	n_{ec}, highest weight
$\texttt{Ala}_8$, $\alpha_L \rightleftharpoons \beta$	772420	870	63
$\texttt{Ala}_8$, $\beta \rightleftharpoons \alpha_R$	772420	865	450
$\texttt{Ala}_8$, $\alpha_L \rightleftharpoons \alpha_R$	772420	1016	71
`Ras p21`, active⇌inactive	47404	2252	n/a

We have also used the algorithm in such a way as to provide an approximate result for the shortest path, with Δ being small enough so that at least the highest edge weight along the shortest path was unambiguously identified. The highest edge weight is biologically the most interesting one, as it provides the molecular structure corresponding to the bottleneck of the transition. The results shown in the fourth column of Table 1 are very encouraging. The numbers of edge weights required are three to four orders of magnitude less than $|E|$. With these savings, the times required for the best-path calculations are reduced from several CPU years to a few CPU days.

3.2 Minimum Cuts

When the inverse mean residence times are used as edge weights in a transition network, the minimum (s,t)-cut is of special relevance as it yields the set of edges corresponding to the slowest, or rate-limiting, part of the transition connecting vertices s and t, also known as *transition state ensemble* [3].

We have computed the minimum cut for the `Ras p21` transition network using Algorithm 1. For the computation of a minimum (s,t)-cut

with given edge weights the algorithm of Nagamochi and Ibaraki was employed. [2]. This minimum cut, which consists of 174 edges, required the computation of 1092 out of a total of $|E| = 47404$ edges. When choosing Δ such that only the highest-weighted edge was computed, $n_{ec} = 805$ edge computations were required, thus reducing the required CPU time to about 5% compared to the trivial solution.

4 Conclusions

In the present paper we have investigated the problem that an optimum solution $\mathrm{OPT}(G)$ needs to be computed for some edge weighted graph G, where initially the weights are not available and can only be obtained at high cost. This is different from the usual setting where complete information is given and the fastest optimization algorithm is sought for.

We have presented a serial and parallel version of a simple heuristic approach and shown that it is very successful for analyzing molecular dynamics using transition networks and that only a very small number of the edge weights need to be known exactly. The approach has reduced the computer time necessary to perform the graph optimization from several CPU years to a few days, which facilitates calculations that would otherwise be out of reach.

These results suggest that graph optimizations in the case where edge weights are not (fully) known is a worthwhile field for further research that may benefit many application areas.

Acknowledgements Frank Noé kindly acknowledges funding from the DFG center Matheon and Landesstiftung Baden-Württemberg

References

1. E. Dijkstra. A note on two problems in connexion with graphs. *Num. Math.*, 1:269–271, 1959.
2. H. Nagamochi and T. Ibaraki. Computing edge connectivity in multigraphs and capacitated graphs. *SIAM Journal on Discrete Mathematics*, 5:54–66, 1992.
3. F. Noé and D. Krachtus and J. C. Smith and S. Fischer. Transition networks for the comprehensive characterization of complex conformational change in proteins. *J. Chem. Theory and Comput.*, 2:840–857, 2006.
4. F. Noé, M. Oswald, G. Reinelt, S. Fischer, and J. C. Smith. Computing Best Transition Pathways in High-Dimensional Dynamical Systems: Application to the $\alpha_L \leftrightharpoons \beta \leftrightharpoons \alpha_R$ Transitions in Octaalanine. *Multiscale Model. Sim.*, 5:393–419, 2006.

Configuration of Order-Driven Planning Policies

Thomas Volling and Thomas S. Spengler

Technische Universität Braunschweig, Lehrstuhl für Produktion und Logistik, Katharinenstr. 3, 38106 Braunschweig
{t.volling|t.spengler}@tu-bs.de

Summary. The intend of this paper is to illustrate the need for complementing the conceptual development of decision models with a structured configuration process. The objective of which is to adjust normative models to the requirements of decision situations that are either multi-criterial or characterized by an open decision field. The argumentation is illustrated for the case of order-driven planning.

1 Introduction

In implementing mass customization, most companies rely on demand driven order fulfillment concepts. When this is done, product configuration is being postponed until the placement of customer orders whilst component production and procurement are being executed based on forecasts. This strategy, commonly denominated as build-to-order (BTO), allows companies to set up order fulfillment systems that are characterized by a balanced mix of efficiency (i.e. scale effects, standardized processes, a high quality) and flexibility (i.e. short lead times, large variability). The major organizational challenge in implementing BTO lies in coping with reduced decoupling mechanisms against the variability and dynamics of the market. With real-world production systems being limited in terms of flexibility, the synchronized adjustment of capacity with the volatility of the market is not viable. Instead, control concepts are needed, to match the supply of resources with the demand for products. The associated planning tasks are referred to as order-driven planning.

Requirements regarding order-driven planning systems are quite diverse. While the demand side is characterized by a high number of customer requests and short response time expectations, the supply

side requires the full complexity of the decision situation to be taken into consideration, while being less demanding with respect to response time. It thus seems to be reasonable to conceptualize order-driven planning as a hierarchical system, comprising distinct modules for order promising (OP) and master production scheduling (MPS) [2]. This approach is likewise reflected by the architecture of most state-of-the-art advanced planning systems.

When decomposing order-driven planning, issues arise concerning the coordination of the dynamic interaction between the planning modules. Given conceptual models, coordination can be achieved by setting the parameters of these models, such that the overall performance is optimized. The associated process will be denominated as configuration. This process is, however, hampered by the fact, that the performance is subject to stochastic influences, because the arrival and configuration of customer orders is not known in advance. The aim of the paper is to illustrate the need to complement the conceptual development of decision models with a configuration process. The objective of which is to adjust normative models to the requirements of a particular decision situation. The approach is illustrated for order-driven planning.

2 Conceptual Framework

In order-driven planning two planning functions can be distinguished. These are the customer interaction (i.e. the quotation of customer requests considering capacities available) and the transformation of these quoted requests into production plans, while taking into account input from the subordinate planning (i.e. the aggregate resource availability). Spengler et al. [2] identify two planning modules, being OP and MPS. The resulting framework is depicted in Figure 1.

A conceptual approach for the development of mathematical programs for order-driven planning is given by [2]. These models have also been evaluated using discrete-event simulation [3]. Accordingly, the proposed approach can be used to significantly improve the performance of order-driven planning systems. Based on these contributions we will in the following present an approach to configure the models up the characteristics of a particular industry setting. In line with this objective, the description of the models will be restricted to their structural basics.

OP seeks to determine due dates for specified customer requests. Due to response time requirements, each order is processed individually (i.e. real-time approach). This yields the objective function given in (1).

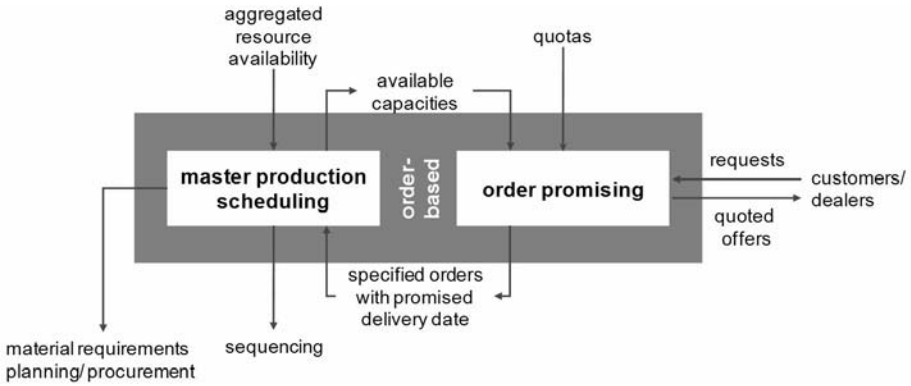


Fig. 1. Framework for order-driven planning

The objective is to minimize time-dependent costs of assigning an order i to a period t within the planning horizon $[t, t+T_{max}]$. To this end, the costs of the assignment $c_{i\tau}$ reflect customer preferences. The assignment is represented by the binary variables $x_{i\tau}$.

$$\min \sum_{\tau=1}^{t+T^{max}} c_{i\tau} \cdot x_{i\tau} \tag{1}$$

The objective of MPS is, in the following, to determine production periods for those orders $i \in \Psi$ that have been quoted for production in the planning horizon $[t, t+T-1]$. We therefore differentiate three terms of the objective function. The first term is set up to incorporate leveling aspects. In doing so, we minimize shortfalls $ctp^{-}_{r\tau}$ to the lower level of the targeted capacity utilization of resource r ($r \in \Omega$) in period τ and weight them using the function $P^{leveling}_{r\tau}(\cdot)$. The objective of the second term is to maximize the available capacity $ctp^{+}_{r\tau}$ weighted by the function $P^{service}_{r\tau}(\cdot)$ as a measure to assess the ability to service new orders. In addition to that, we incorporate costs $\overline{c}_{i\tau}$ that are due to deviations between the due date assigned by OP and that quoted by the MPS procedure (e.g. holding costs). The assignment is modeled using the binary variables $\overline{x}_{i\tau}$. A central feature of the modeling approach is the segmented objective function, which differentiates two intervals marked by the parameter k. A more detailed description is given by [3]. This yields the following objective function:

$$\min \sum_{\tau=t}^{t+k} \sum_{r\in\Omega} P^{leveling}_{r\tau}(ctp^{-}_{r\tau}) - \sum_{\tau=t+k+1}^{t+T-1} \sum_{r\in\Omega} P^{service}_{r\tau}(ctp^{+}_{r\tau}) + \sum_{\tau=t}^{t+T-1} \sum_{i\in\Psi} \overline{c}_{i\tau} \cdot \overline{x}_{i\tau} \tag{2}$$

The implementation of the objective function requires the economical assessment of the particular terms. The explicit determination of monetary consequences often raises difficulties in real-world settings, since data is not available in the sufficient quality and quantity or because the computational complexity does not allow for an explicit determination. This essentially leads to a multi-criteria decision situation, which requires the specification of additional parameters in order to identify preferred solutions. In addition to that, the decision situation is subject to a stochastic demand process. Accordingly, the informational basis (i.e. the customer orders placed) evolves in time. This constitutes an open decision field [2, pp. 39]. In order to cope with this fact, [3] argue for MPS to be executed based on rolling horizons. As a consequence, it is not the single execution that matters but the aggregated performance of the planning system which results from the repeated interaction of the models. This fact, however, hampers the interpretability of the parameters of the objective function. Thus there is a need for a structured process to identify beneficial parameter combinations for a particular setting. We will provide a more formal analysis in the following.

3 Configuration of Order-Driven Planning

Generally speaking, two methods of setting model parameters can be distinguished. In many situations it is not possible to quantify or observe all the measures (model parameters) necessary to set up an adequate model. Different techniques exist which allow for the determination of these parameters, such that a match can be obtained between the model and the underlying real-world phenomenon. The objective of this is to minimize the explanatory error – typically using past data. The associated process is commonly referred to as calibration and is most suitable for explanatory and forecasting models respectively (e.g. time series forecasting).

A different situation can be found when considering the task of setting parameters in decision models. In contrast to the explanatory application of models, the aim in this case is to provide normative decision support. Accordingly, parameters need to be identified, such that the model generates solutions in line with the decision maker's preferences. The most prominent case of this situation is that of multi-criteria decision making. In this domain, Hanne [1] provides an approach based on techniques from artificial intelligence. A second one, which is under-

represented in literature, regards decision situations with open decision field. We will denote the associated tasks as configuration.

As the performance of planning schemes in the presence of on open decision field can only be assessed on an aggregated basis, the configuration process requires the planning modules to be run multiple times. The configuration task therefore constitutes a meta-problem subordinate to the original decision model(s). The complete setting can be summarized as shown in Figure 2. The decision model is executed with a certain set of parameter values against specific environmental conditions. The resulting performance is fed back into the configuration module, which evaluates the performance with respect to the preferences of the decision maker. The process is re-iterated until defined stopping conditions are obtained (e.g. solution quality, run time). This leads to

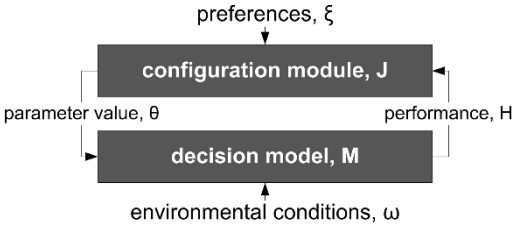


Fig. 2. Framework for the configuration process

the following formal problem definition. The objective of the configuration procedure is to identify the deterministic parameter (vector) θ^* out of all feasible (vectors) $\theta \in \Theta$, that optimizes the performance H of the planning system M. If there is more than one performance measure, they need to be aggregated into the scalar function J by means of the preference vector ξ, prior to the analysis. The performance of the planning system might, furthermore, be influenced by (stochastic) environmental conditions. If the complexity of the setting does not allow for an analytical solution, J needs to be estimated adequately. A promising solution approach lies in the application of simulation-based analysis. Doing so, J is commonly given by the expected mean based on the sample ω (simulation replications). The configuration calculus can thus be given as:

$$\theta_\xi^* = argopt_{\theta \in \Theta} J_\xi(\theta) = J_\xi(E[H(M(\theta), \omega)]) \tag{3}$$

As elaborated upon above, the decision situation of order-driven planning requires normative modeling and exhibits both multiple objectives and an open decision field. In order to provide decision support for particular settings, the planning system therefore needs to be configured. Given the objective functions (1) and (2), the configuration task regards the definition of the functions $P_{r\tau}^{leveling}(\cdot)$ and $P_{r\tau}^{service}(\cdot)$. [3] provide evidence, that when using additive weighting for both terms, the associated scalar weights can be used to control the aggregated performance. The process to configure decision models for order-driven planning therefore comes down to the determination of scalar weights of the first and the second term of the MPS objective function such that the performance is optimized for a given demand pattern and resource characteristics.

4 Conclusions

The decision situation of order-driven planning is characterized by multiple criteria and an open decision field. Mathematical models for decision support therefore need to be adjusted to the requirements of particular situations. We refer to the associated process as configuration. Since configuration requires the model(s) to be run multiple times, efficient search routines to identify beneficial parameter combinations are essential. Corresponding techniques can be found in the realm of simulation optimization. Future research needs to be done to illustrate the potentials of this approach.

References

1. T. Hanne. *Intelligent strategies for meta multiple criteria decision making.* Number 33. Kluwer Academic, Boston, 2001.
2. T. Spengler, T. Volling, and S. Rehkopf. Taktisch-operative Planung bei variantenreicher Serienfertigung. In M. Jacquemin, R. Pibernik, and E. Sucky, editors, *Quantitative Methoden der Logistik und des Supply Chain Management*, pages 183–210. Verlag Dr. Kovac, 2006.
3. T. Volling and T. Spengler. Order-driven planning in build-to-order scenarios. *Tagungsband Wirtschaftsinformatik 2007*, pages 183–210, 2007.

Part XIX

Supply Chain Management and Traffic

When Periodic Timetables Are Suboptimal*

Ralf Borndörfer[1] and Christian Liebchen[2]

[1] Konrad-Zuse-Zentrum für Informationstechnik Berlin, Takustr. 7, 14195 Berlin-Dahlem, Germany. borndoerfer@zib.de
[2] Technische Universität Berlin, Institut für Mathematik, Str. des 17. Juni 136, 10623 Berlin, Germany. liebchen@math.tu-berlin.de

Summary. "The timetable is the essence of the service offered by any provider of public transport" (Jonathan Tyler, CASPT 2006). Indeed, the timetable has a major impact on both operating costs and on passenger comfort. Most European agglomerations and railways use periodic timetables in which operation repeats in regular intervals. In contrast, many North and South American municipalities use trip timetables in which the vehicle trips are scheduled individually subject to frequency constraints. We compare these two strategies with respect to vehicle operation costs. It turns out that for short time horizons, periodic timetabling can be suboptimal; for sufficiently long time horizons, however, periodic timetabling can always be done 'in an optimal way'.

1 The Timetabling Problem

The construction of the timetable is perhaps the most important scheduling activity of a railway or public transport company. It has a major impact on operating costs and on passenger comfort. The problem has been extensively covered in the operations research literature, see [2] for a recent survey. There are two main timetabling strategies that differ w.r.t. to structural dependencies between individual trips. In a periodic timetable, there is a fixed time interval between two trips; if a single trip is scheduled on a directed line, all other trips of this line are determined. In contrast, in a trip timetable, each trip is scheduled individually, subject to frequency constraints. Stipulating appropriate constraints, a trip timetable can be forced to become periodic, or, to put it the other way round, trip timetables feature more degrees of freedom than periodic ones. We investigate in this paper the question

* Supported by the DFG Research Center MATHEON (http://www.matheon.de).

whether this freedom can be used to lower operation costs in terms of numbers of vehicles. Of course, such improvements (if any) come at the price of diminishing the regularity of the timetable.

We consider the timetabling problem for a single, bidirectional line between two stations A and B. The line is operated by homogenous vehicles with running times t_{ab} and t_{ba} in directions $A \to B$ and $B \to A$, respectively (these include minimum turnaround times in the respective terminus stations). We want to construct a timetable that covers N time periods of length T with a trip frequency of f vehicles per time period, and such that the minimum headway between two consecutive trips is at least ℓ and at most u. We assume that f divides T and call T/f the *period time* of the timetable (to be constructed). We further assume $\ell \leq T/f \leq u \leq T$ and that all mentioned numbers are positive integers except for ℓ, which is supposed to be a non-negative integer. The timetable that we want to construct involves $m := N \cdot f$ departures at station A, that we denote by $U = \{u_1, \ldots, u_m\}$, and the same number of departures at station B, that we denote by $V = \{v_1, \ldots, v_m\}$; let $U \cup V$ be the set of all these departure events. A *timetable* is a function $t : U \cup V \mapsto \mathbb{Z}$ that maps departures to times such that the following conditions hold:

$$
\begin{array}{lll}
\text{(i)} & t(u_i) \leq t(u_{i+1}) & i = 1, \ldots, m-1 \\
& t(v_i) \leq t(v_{i+1}) & i = 1, \ldots, m-1 \\
\text{(ii)} & \lfloor (i-1)/f \rfloor T \leq t(u_i) < (\lfloor (i-1)/f \rfloor + 1)T & i = 1, \ldots, m-1 \\
& \lfloor (i-1)/f \rfloor T \leq t(v_i) < (\lfloor (i-1)/f \rfloor + 1)T & i = 1, \ldots, m-1 \\
\text{(iii)} & \ell \leq t(u_{i+1}) - t(u_i) \leq u, & i = 1, \ldots, m-1 \\
& \ell \leq t(v_{i+1}) - t(v_i) \leq u, & i = 1, \ldots, m-1.
\end{array}
$$

Constraints (i) ensure that the departure times at both stations ascend in time, (ii) guarantees f departures in each period at each station, and (iii) enforce a minimum and maximum headway of ℓ and u between two consecutive departures of trips. A timetable is a *periodic timetable* if condition (iii) is replaced by

$$
\text{(iii}') \quad t(u_{i+1}) - t(u_i) = t(v_{i+1}) - t(v_i) = T/f, \qquad i = 1, \ldots, m-1,
$$

otherwise it is a *trip timetable*. Note that a timetable can be forced to be periodic by stipulating $\ell = T/f = u$. The *timetabling problem* is to determine a feasible timetable that can be operated with a minium

number of vehicles[3]. We remark that our problem definition deliberately omits technical constraints, such as passing sidings, in order to focus upon purely structural implications.

2 Periodic vs. Trip Timetables

Lemma 1. *Consider a public transport line between stations A and B with running times t_{ab} and t_{ba} which include the minimum turnaround times in the respective terminus stations, such that $t_{ab}+t_{ba}$ is an integer multiple of T. Then, operating this line for a time duration of at least $N \cdot T > t_{ab} + t_{ba}$ requires at least*

$$Z := \frac{t_{ab} + t_{ba}}{T/f} \tag{1}$$

vehicles in an arbitrary timetable.

Proof. At least f vehicles have to be scheduled in each of the first Z/f time periods until the first vehicle can be reused for a second trip in the same direction. □

Lemma 2. *Consider a public transport line between stations A and B with running times t_{ab} and t_{ba} (again including minimum turnaround times). Operating this line at shorter running times $t'_{ab} \leq t_{ab}$ and $t'_{ba} \leq t_{ba}$ does not increase the number of vehicles that are required for operation in the respectively best arbitrary timetables for this line.*

Proof. For the optimal timetables for running times t_{ab} and t_{ba} there exist timetables with running times t'_{ab} and t'_{ba} that can be operated at the same number of vehicles. In fact, add a turnaround waiting time at terminus station B of $t_{ab} - t'_{ab}$, and a waiting time of $t_{ba} - t'_{ba}$ at station A. □

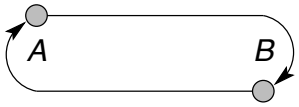


Fig. 1. The so-called PESP-graph (see, e.g., [2]) for the situation of the periodic vehicle circulation as it is considered in Prop. 1

[3] This definition is made in our context; there are other types of timetabling problems in the literature.

Proposition 1. *Consider a public transport line between stations A and B with a period time $\frac{T}{f}$ and running times t_{ab} and t_{ba} (again including minimum turnaround times in the terminus stations) such that $t_{ab} + t_{ba} < NT - T/f$. Then, any periodic timetable requires at least*

$$Z_0 := \left\lceil \frac{t_{ab} + t_{ba}}{T/f} \right\rceil \tag{2}$$

vehicles for operation. Moreover, any periodic timetable for this line can be operated with at most $Z_0 + 1$ vehicles. Indeed, there exist periodic timetables that can be operated with Z_0 vehicles.

Proof. Using the cycle inequalities due to Odijk [4], it had been observed by Nachtigall [3] that there exists some appropriate $\varepsilon > 0$ such that the following general bounds on the number Z of vehicles are valid for all periodic timetables, and tight for some timetables:

$$Z_0 := \left\lceil \frac{t_{ab} + t_{ba}}{T/f} \right\rceil \leq Z \leq \left\lfloor \frac{t_{ab} + \frac{T}{f} - \varepsilon + t_{ba} + \frac{T}{f} - \varepsilon}{T/f} \right\rfloor \leq Z_0 + 1. \tag{3}$$

□

Lemma 3. *Consider a public transport line between stations A and B with running times t_{ab} and t_{ba} which once more include the minimum turnaround times in the respective terminus stations. Then, operating this line for at least a time duration of $N \cdot T > t_{ab} + t_{ba}$ requires at least*

$$Z_1 := \left\lfloor \frac{t_{ab} + t_{ba}}{T} \right\rfloor \cdot f \tag{4}$$

vehicles in an arbitrary timetable.

Theorem 1. *For each public transportation line with running times t_{ab} and t_{ba} (including minimum turnaround times), there exists a number $N_0 \in \mathbb{N}$ such that operating the line for a time duration of at least $N_0 \cdot T$ requires at least Z_0 vehicles for operation. In other words, for sufficiently long time horizons, the minimum number of vehicles needed to operate a trip timetable is equal to the minimum number of vehicles needed to operate a periodic timetable.*

Proof. In a time duration of $N \cdot T$, for the two directions of the line together there must be scheduled at least $2Nf$ trips. In turn, one vehicle can cover no more than $\left\lceil \frac{2NT}{t_{ab}+t_{ba}} \right\rceil$ of these trips.

Now, choose N_0 such that $\frac{2N_0T}{t_{ab}+t_{ba}}$ becomes integer. Then, the number of required vehicles is bounded from below by

$$\frac{2N_0 f}{\frac{2N_0 T}{t_{ab}+t_{ba}}} = \frac{f}{T} \cdot (t_{ab} + t_{ba}) = \frac{t_{ab} + t_{ba}}{T/f}. \tag{5}$$

Since we must only consider integer quantities of vehicles, the claim follows. □

3 Example

Let the balancing intervalequal $T = 60$ minutes, and let the number of required trips within this interval be $f = 3$. We consider a parameterized one-way running time $t_{ab} = t_{ba} = 60 - c$, which includes the minimum turnaround times, for $c \in \{1, 2, \ldots, 10\}$.

First, observe that whenever $c < 10$, then the number of vehicles that are required in the best periodic timetables equals

$$Z_0 = \left\lceil \frac{t_{ab} + t_{ba}}{T/f} \right\rceil = \left\lceil \frac{120 - 2c}{60/3} \right\rceil = \left\lceil \frac{120 - 2c}{20} \right\rceil = \left\lceil \frac{60 - c}{10} \right\rceil = 6. \tag{6}$$

Second, it can be verified that in any trip timetable, we need at least five vehicles for operating this line over at least three hours. Last, recall from the proof of Theorem 1 that after at least $100-2c$ hours we can be sure to also require six vehicles when scheduling each trip individually. Yet, in this example we will show that, for certain running times, as early as after at least six hours we are sure to need the sixth vehicle also in any trip timetable.

The first simple observation is that within each hour, when considering both directions of the line together, there must be six trips in the schedule. Hence, in order to need only five vehicles, there must be one vehicle within each hour that covers two of these six trips. Of course, these two trips must be in opposite direction.

Now comes the key observation: If some fixed vehicle covers two trips in hour X, then it cannot cover two trips in any of the hours $\{X + 1, \ldots, X + \lfloor \frac{T}{c} \rfloor - 2\}$. As a consequence, if the one-way running time was $t_{ab} = 52$, and thus $c = 8$, then no vehicle can cover two trips in two of the hours $\{1, \ldots, 6\}$, because of $X + \lfloor \frac{T}{c} \rfloor - 2 = 1 + 7 - 2 = 6$. Hence, in the sixth hour of operation, the latest, a sixth vehicle has to be put into operation, cf. Fig. 2.

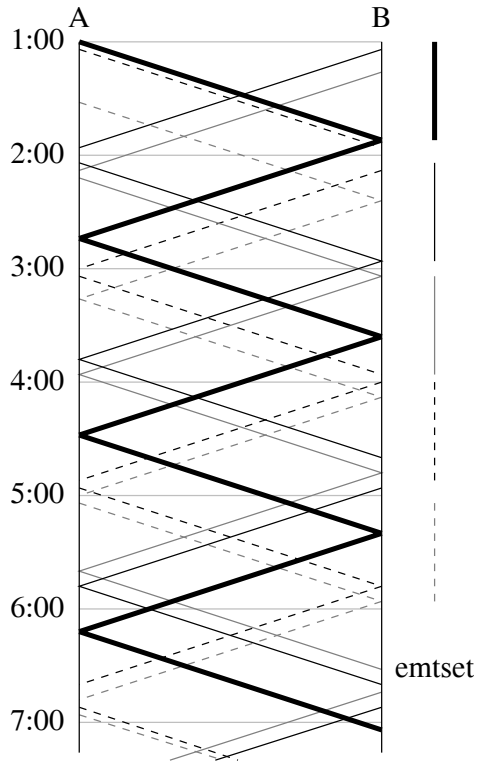


Fig. 2. A timetable for $t_{ab} = t_{ba} = 52$, that uses only five vehicles. In the first five hours, there are three departures from each terminus station. But in the sixth hour, there are only two departures from B, which makes this timetable infeasible for $N = 6$. The lines on the right indicate the vehicle that covers two trips in the corresponding hour

References

1. Christian Liebchen. Fahrplanoptimierung im Personenverkehr—Muss es immer ITF sein? *Eisenbahntechnische Rundschau*, 54(11):689–702, 2005. In German.
2. Christian Liebchen. *Periodic Timetable Optimization in Public Transport.* Ph.D.thesis, Technische Universität Berlin, 2006. dissertation.de—Verlag im Internet.
3. Karl Nachtigall. *Periodic Network Optimization and Fixed Interval Timetables.* Habilitation thesis, Universität Hildesheim, 1998.
4. Michiel A. Odijk. A constraint generation algorithm for the construction of periodic railway timetables. *Transportation Research B*, 30(6):455–464, 1996.

Acceleration of the A*-Algorithm for the Shortest Path Problem in Digital Road Maps

Felix Hahne, Curt Nowak, and Klaus Ambrosi

Institut für Betriebswirtschaft und Wirtschaftsinformatik
(www.bwl.uni-hildesheim.de)
Universität Hildesheim, Marienburger Platz 22, 31141 Hildesheim, Germany

Summary. Using the A*-algorithm to solve point-to-point-shortest path problems, the number of iterations depends on the quality of the estimator for the remaining distance to the target. In digital maps of real road networks, iterations can be saved by using a better estimator than the Euclidian estimator. An approach is to integrate Segmentation Lines (*SegLine*) into the map modelling large obstacles. An auxiliary graph is constructed using the SegLines wherein a shortest path is calculated yielding a better estimate. Some computational results are presented for a dynamic version of this approach.

1 Speeding up the A*-Algorithm

The A*-algorithm ([9], [10] and [15]) is among the most popular algorithms to solve point-to-point problems from a starting point s to a target t. Storing its results in a search tree, the number of iterations taken is closely correlated to the amount of memory consumed.

Applied to huge graphs, like digital maps of real road networks, the search tree may become very large. This is a problem in environments where resources are limited, e.g. mobile car navigation-systems with limited memory and map data loading over small bandwidth-interface.

Different preprocessing approaches have been researched to reduce the number of iterations taken and to limit the size of the search tree:

- Upper bounds ('radius', 'reach') for arcs to exclude them from processing when they cannot be part of the optimal path ([2] and [6]).
- Simplification of the map, e.g. subsuming into hierarchical levels with a lower number of nodes and arcs ([17], [18]).
- Pruning the map ([19]) or tree ([14]) using geometric information.

A more direct way focusses on the estimator function h_t used by the A*-algorithm. Function h_t estimates the distance of the currently scanned node to the target. During the iteration process, the next item selected from the list of possible items will be the one minimizing d_s+h_t, with d_s being the length of the best known path from s. Thus the area searched is driven into the direction of t.

An estimator never over-estimating the real distance to t, thus satisfying Bellmann's condition, is called dual feasible or consistent. Using such an estimator, the shortest path is found ([12], [16]). By intentionally over-estimating ('overdoing') the real distance the algorithm becomes a greedy heuristic ([7], [11]).

In digital maps of real road networks, the standard dual feasible estimator is the Euclidian (air-line) distance. As shown in [3] and [9], the A*-algorithm makes optimal use of the information contained within this estimator. Empirical analysis on such maps in [7] and [11] show the number of iterations taken (closely correlated: the amount of memory used) by the A*-algorithm decreases roughly by the factor of four compared to Dijkstra's algorithm.

Better (tighter) estimators lead to fewer iterations: if E_1 yields constantly better results than E_2, then the nodes scanned by the A*-algorithm using E_1 are a subset of those scanned using E_2 ([4]).

Using a 'perfect' estimator P always yielding the exact real distance, you only have to choose the best successor of the current node; the search tree is nearly limited to the shortest path itself ([8], [4]). In road networks with an average node degree of around three, the number of iterations using P goes down to 5% compared using the Euclidian estimator ([8]). However, calculating P is equal to solving the original problem itself.

So there is a trade-off between runtime and memory: the more you invest in the calculation of the estimator, the less iterations and memory the A*-algorithm consumes. When using preprocessed information, there is a second trade-off: the preprocessed data must not be too large or need too much time to be retrieved.

2 Better Estimators for the A*-Algorithm

2.1 Computing Lower Bounds from Landmarks

A newer approach ([4], [5], and [13]) is the ALT-algorithm ('A* with landmarks and triangle inequality'). It selects a small number of nodes as landmarks and precomputes the shortest paths to all other nodes.

Lower bounds for the distance d_{nt} between node n and t are derived from the triangle inequality: $d_{nt} \geq d_{L_1 t} - d_{L_1 n}$ and $d_{nt} \geq d_{nL_2} - d_{tL_2}$. The tightest lower bound is the maximum over all such bounds.

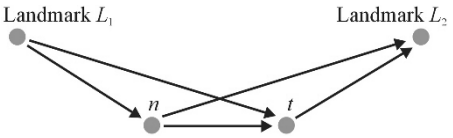


Fig. 1. Using Landsmarks to derive lower bounds

2.2 Modelling Obstacles Through Segmentation Lines

The SegLine-approach ([1]) incorporates obstacles like mountains or pedestrian areas—areas containing no arcs used by the shortest path algorithm. The idea is to describe their extent by additional SegLines, which must not be crossed by Euclidian estimator lines. Fig. 2 shows several SegLines (thick grey lines).

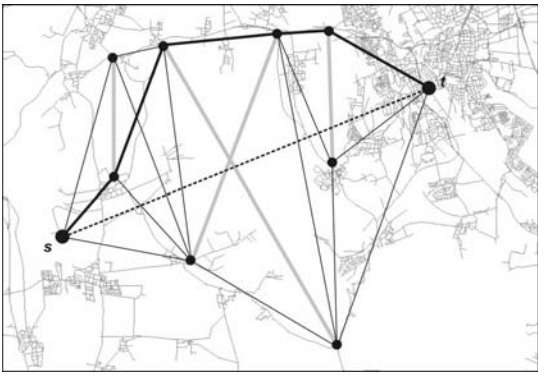


Fig. 2. Modelling obstacles with segmentation lines

The improved estimator for the distance between s and t is the shortest path in an auxiliary graph which is constructed as follows:

- The set of nodes consists of all endpoints of segmentations lines and the two nodes s and t (black dots in Fig. 2).
- The set of arcs comprises all Euclidian lines between nodes not crossing SeLines (solid black lines in Fig. 2); all arc-lengths are Euclidian.

The resulting shortest path (thick black lines) is a better estimator than the Euclidian estimator (dotted line), but still a lower bound.

Implemented naively, the complexity is $O(m^2)$ with m the number of SegLines. The following preprocessing is suggested in [1]:

- Determine for each node of the original graph the set of "'visible"' endpoints, i.e. reachable by Euclidian lines without crossing SegLines (to be performed just once).
- In the auxiliary graph, calculate the shortest paths between all endpoints of SegLines and the target (to be redone for each target).

Then checking the endpoints visible from the current node determines the estimator in $O(m)$.

In [1] a main roads map of Switzerland with some 13,000 nodes, 20,000 arcs and 142 SegLines (ratio: 141 arcs per SegLine) was used. The average decrease in number of scanned nodes was approx. 25%.

3 Using Segmentation Lines Without Preprocessing

Both approaches require a complete workthrough of the nodes of the map plus the storage of several attributes. Especially the landmark apporach is vulnerable to changes in the map, which require costy updates of the shortest paths to the landmarks.

3.1 The Dynamic Segmentation Line-Estimator

The following approach is a dynamic version of the one described in section 2.2 requiring no preprocessing. The differences are:

- The SegLines used for the auxiliary graph are only those cut by the Euclidian line between the current item and the target. E.g. in Fig. 2 the leftmost SegLine would be neglected. The determination of this subset is done during runtime.
- The shortest path in the auxiliary graph is calculated online as well using the A*-algorithm with the Euclidian estimator.

The result is a small auxiliary graph with just relevant SegLines.

3.2 First Computational Results

Empirical tests were done on a map of the city of Hildesheim (6,200 nodes, 9,000 arcs). 870 shortest paths are calculated between 30 selected

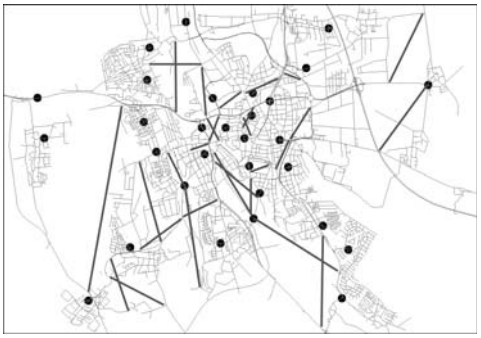


Fig. 3. Benchmark Hi30 with 30 segmentation lines

nodes evenly spread on the map (Benchmark "'Hi30"', [7]). Sets of 10, 20, 30 and 40 manually selected SegLines are added. Acceleration is measured comparing the number of iterations to those taken by the A*-algorithm with Euclidian estimator:

Table 1. Computational Results

#Segmentation lines	Average reduction
10	7.4%
20	10.1%
...	...
60	22.3%

The average reduction is, lower than in [1], as a higher ratio of SegLines is used (max. ratio: 226 arcs per SegLine) and not all SegLines are used. Running time is higher, but only a little extra memory is used.

4 Conclusion and Future Works

The concept of only using cut SegLines seems to extract most of the information, no preprocessing is necessary and very little memory used. The approach is quite robust concerning map changes, as SegLines only have to be corrected when arcs are added which are cut by SegLines.

Further research goes into developing criteria for a good SegLine, into automatically finding them and into statistic evaluati on of their use within the estimator.

To speed up, one idea is not to use the improved estimator in every iteration. Results from [8] show that even an estimator with erratic behaviour can greatly help reducing the number of iterations.

References

1. Dubois N., Semet F. (1995) Estimation and determination of shortest path length in a road network with obstacles. EJOR 83:105-116.
2. Ertl, G. (1998) Shortest path calculations in large road networks, OR Spektrum 20:15-20.
3. Gelperin, D. (1977) On the Optimality of A*, AI 8:69-76.
4. Goldberg, A.V. (2006) Point-to-Point Shortest Path Algorithms with Preprocessing, Technical Report, Microsoft Research.
5. Goldberg, A.V., Harrelson, C. (2004) Computing the Shortest Path: A* Search Meets Graph Theory, MSR-TR-2004-24, MS Research.
6. Gutman, R. (2004) Reach-based Routing: A New Approach to Shortest Path Algorithms Optimized for Road Networks, Proc. 6th International Workshop on Algorithm Engineering and Experiments, 100-111.
7. Hahne, F. (2000) Kürzeste und schnellste Wege in digitalen Straßenkarten, Dissertation, Universität Hildesheim.
8. Hahne, F. (2005) Analyse der Beschleunigung des A*-Verfahrens durch verbesserte Schätzer für die Restdistanz, OR Proceedings 2005.
9. Hart, P.E., Nilson, N.J., Raphael, B. (1968) A formal basis for the heuristic determination of minimal cost paths. IEEE Tr. on SSC 4:100-107.
10. Hart, P.E., Nilson, N.J., Raphael, B. (1972) Correction to: 'A formal basis for the heuristic determination of min. cost paths'. Sigart NL 37:28-29.
11. Hasselberg, S. (2000) Some results on heuristical algorithms for shortest path problems in large road networks, Dissertation, Universität Köln.
12. Ikeda T., Hsu M., Imai H. (1994) A fast algorithm for finding better routes by a.i. search techniques, VNIS Conference Proceedings 1994.
13. Klunder, G.A., Post, H.N. (2006) The shortest path problem on large scale real road networks, Networks 48:184-192.
14. Lauther, U. (2004) An Extremely Fast, Exact Algorithm for Finding Shortest Paths in Static Networks with Geographical Background, IfGIprints 22, Institut für Geoinformatik, Universität Münster 219-230.
15. Nilsson, N. (1971) Problem-Solving Methods in Artificial Intelligence. McGraw-Hill, New York.
16. Pijls, W. (2007) Heuristic estimates in shortest path algorithms, Statistica Neerlandica 61:64-74.
17. Sanders, P. Schultes, D. (2005) Highway hierarchies hasten exact shortest path queries, in: Brodal, G.S., Leonardi, S. (eds), Proc. 17th ESA, Lecture Notes in CS 3669, Springer, 568-579.
18. Stausberg, V. (1995) Ein Dekompositionsverfahren zur Berechnung kürzester Wege auf fast-planaren Graphen, Diploma Thesis, Univ. Köln.
19. Wagner, D., Willhalm, T. (2003) Geometric Speed-Up Techniques for Finding Shortest Paths in Large Sparse Graphs, Proc. 11th ESA.

A Modulo Network Simplex Method for Solving Periodic Timetable Optimisation Problems

Karl Nachtigall and Jens Opitz

TU Dresden, Faculty of Transport and Traffic Science, Institute for Logistics and Aviation, Department for Traffic Flow Science
karl.nachtigall@tu-dresden.de, jens.opitz@tu-dresden.de

1 Introduction

In the last 15 years periodic timetable problems had been found much interest in combinatorial optimization. The results presented in [5, 9, 2, 3, 6, 7, 1] are based on a periodic event scheduling model published by Serafini and Ukovich [10].

Periodic event-activity networks allow a flexible modelling of fixed interval timetables in public transport. A lot of practical requirements, like sequencing of trains, safety headway distances and limits for rolling stock can be incorporated into this network theoretical model. In this paper we will focus on the optimisation task to minimise a weighted sum of undesirable slack times, e.g. waiting time for passengers.

Define a *periodic railway system* by a system of lines $\mathcal{L}$ and stations $\mathcal{S}$. Each line $L \in \mathcal{L}$ is understood to be a *transportation chain*, where the trains of L are serving a certain sequence of stations in fixed time intervals of T minutes (see e.g. [11]). If line L serves stations S, then define (L, arr, S) and (L, dep, S) to be the arrival (departure) event of L at S.

A *schedule* assigns *event times* $\pi_i \in \mathbb{R}$ to all events $i = (L, dep, S)$ or $i = (L, arr, S)$. An activity $a : i \to j$ is a time consuming process, which then will consume the amount $x_a := \pi_j - \pi_i$ of time.

A line can be understood as an alternative sequence of

- run activities : $(L, dep, S) \to (L, arr, S')$ and
- stop activities : $(L, arr, S) \to (L, dep, S)$.

Run and stop activities are assigned with time spans $\Delta_a = [\ell_a, u_a]$, where ℓ_a is the minimum running or stopping time and u_a is an upper

bound.[1] A schedule $\boldsymbol{\pi}$ is said to be *feasible*, if $x_a = \pi_j - \pi_i \in \Delta_a$ for all $a : i \to j$. Apart from running and stopping activities, in real world problems there are many other types of constraints arising from operational, safety- or marketing-related restrictions. The most important operational are headway constraints, which separate trains running on the same part of the infrastructure.

Those non-periodic timetable problems are very easy to solve by using shortest path calculations. Fixed interval timetables, where that all departure and arrival events will be repeated periodically, such a simple model is no more pro appropriate. The reasons are manifold: A priori it is not clear between which trains passengers are changing or in which sequence trains are leaving or entering stations. All this can only be decided after if the time ordering of all events is known. A *periodic schedule* assigns periodic *event times* $\pi_i \in \mathbb{R}$ to all events, which will take place at all time points $\pi_i + zT\,(z \in \mathbb{Z})$. The integer multiples z of the period are called *modulo parameter* and code the periodic sequence of all events. The resulting planning problems are known to be NP-hard.

For reasons of simplicity we assume one common period T for the complete system. Different periods for the lines can be handled by using the least common divisor (compare for [4]).

A solution of the periodic timetable problem is defined by $\boldsymbol{\pi} \in \mathbb{R}^n$, which defines for each event i one point of time π_i, such that i will be periodically repeated at all times $\pi_i + zT\,(z \in \mathbb{Z})$. An activity $a : i \to j$ is a time consuming process, which then will consume the amount $x_a := \pi_j - \pi_i$ of time. The process times are given by the *period tension* vector $\mathbf{x} := \Theta^t\boldsymbol{\pi} - \mathbf{z}T$. Define ℓ_a and u_a to be the minimum and maximum allowed process time. Then a timetable $\boldsymbol{\pi}$ is feasible, if and only if $\ell_a \le \pi_j - \pi_i - z_aT \le u_a$.

Lower and upper slack time measures that amount of time for which the tension on this arc may be increased or decreased. Since lower and upper slack times may be exchanged by inverting the direction, the opimisation problem can be defined in terms of lower slack time. The collection of all slack times is given by:

$$\mathbf{y} = \Theta^t\boldsymbol{\pi} - \boldsymbol{\ell} - \mathbf{z}T = \left[\Theta^t\boldsymbol{\pi} - \boldsymbol{\ell}\right]_T$$

Now, the periodic timetable slack problem can be formulated as mixed integer problem

[1] If the runnwing time is fixed, a running activity and the following stop activity can be simply described by one combined constraint.

$$\begin{array}{l}(\Theta\boldsymbol{\omega})^t\,\boldsymbol{\pi} - \boldsymbol{\omega}^t\boldsymbol{\ell} \to \min \\ \boldsymbol{\ell} \le \Theta^t\boldsymbol{\pi} - \mathbf{z}T \le \mathbf{u} \\ \boldsymbol{\pi} \text{ of any sign}\end{array} \tag{1}$$

2 The Periodic Timetable Polyhedron

2.1 Periodic Tensions, Cuts and Flows

Instead of using potential, we may use the associated tension $\mathbf{x} := \Theta^t\boldsymbol{\pi}$, which are characterized by the use of the network matrix Γ in terms of $\Gamma\mathbf{x} = \mathbf{0}$. A *periodic tension* $\mathbf{x}$ fulfills $\Gamma\mathbf{x} \equiv_T \mathbf{0}$. The orthogonal projection of the tension space is known to be the space of all flows ([8]), i.e. it holds $\{\mathbf{x} \mid \Gamma\mathbf{x} = \mathbf{0}\}^{\perp} = \{\boldsymbol{\varphi} \mid \Theta\boldsymbol{\varphi} = \mathbf{0}\}$. In the periodic case, we obtain

$$\{\mathbf{x} \in \mathbb{Z}^m \mid \Gamma\mathbf{x} \equiv_T \mathbf{0}\}^{\perp_T} = \{\boldsymbol{\varphi} \in \mathbb{Z}^m \mid \Theta\boldsymbol{\varphi} \equiv_T \mathbf{0}\} \tag{2}$$

Theorem 1. *Let $\mathcal{Q} \neq \emptyset$. Then $\boldsymbol{\vartheta}^t\boldsymbol{\pi} - \mathbf{f}^t\mathbf{z} \ge r$ can only be a valid inequality for the polyhedron*

$$\mathcal{Q} := conv.hull\left(\left\{\begin{pmatrix}\boldsymbol{\pi}\\ \mathbf{z}\end{pmatrix} \;\middle|\; \boldsymbol{\ell} \le \Theta^t\boldsymbol{\pi} - T\mathbf{z} \le \mathbf{u}; vecz \in \mathbb{Z}^m; \boldsymbol{\pi} \in \mathbb{R}^n\right\}\right)$$

with $\boldsymbol{\vartheta}^t\boldsymbol{\pi}^{(0)} - \mathbf{f}^t\mathbf{z}^{(0)} = r$ for at least one $\begin{pmatrix}\boldsymbol{\pi}^{(0)}\\ \mathbf{z}^{(0)}\end{pmatrix} \in \mathcal{Q}$, if and only if $\mathbf{f}$ is a flow with bilance $\boldsymbol{\vartheta}$, i.e. it holds $T\boldsymbol{\vartheta} = \Theta\mathbf{f}$ and

$$Tr = \min\left\{\mathbf{f}^t\mathbf{x} \;\middle|\; x \in \mathcal{X}\right\}$$

2.2 A Modulo-Simplex Method

The integrality of the modulo parameter $\mathbf{z}$ makes the problem hard. For this reason we will eliminate those variables and keep them implicitly in the model by using modulo calculations. The modulo simplex method explores the extreme points of this polyhedron by spanning trees, which induce basic solutions.

By using $\mathbf{b} :\equiv_T -\Gamma\boldsymbol{\ell}$ and $\boldsymbol{\delta} := \mathbf{u} - \boldsymbol{\ell}$, the periodic slack space is defined by

$$\mathcal{Y} := \{\mathbf{y} \in \mathbb{Z}^m \mid \Gamma\mathbf{y} \equiv_T \mathbf{b}; \mathbf{0}\mathbf{y} \le \boldsymbol{\delta}\}$$

and the optimization task is to determine $\min\left\{\boldsymbol{\omega}^t\mathbf{y} \;\middle|\; \mathbf{y} \in \mathcal{Y}\right\}$. The tree and co-tree arcs of the underlying spanning tree split the network matrix $\Gamma = [N_T, E_T^{co}]$ into its basic (= co-tree) and non-basic (= tree)

components. Therefore a periodic basic solution is given by $\begin{pmatrix} \mathbf{y}_{\mathcal{T}} \\ \mathbf{y}_{\mathcal{T}}^{co} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{b} \end{pmatrix}$, which is feasible if $\mathbf{b} \leq \boldsymbol{\delta}$. Any period tension $\mathbf{x}$ (with $\Gamma\mathbf{x} \equiv_T \mathbf{0}$) leads to a new solution $\mathbf{y}' := [\mathbf{y} + \mathbf{x}]_T = \mathbf{y} + \mathbf{x} - \mathbf{z}'T$ of $\Gamma\mathbf{y}' \equiv_T \mathbf{b}$ and stays feasible, if $\mathbf{y}' \leq \boldsymbol{\delta}$.

A basic exchange can be described by exchanging a leaving co-tree arc a_l with an entering tree arc a_e, which belong to the uniquely determined co-tree cycle of the actual tree. The resulting cut $\boldsymbol{\eta}^{(a_l,a_e)}$ is given by adjoining the leaving tree component to the a_l- associated column of N. This vector $\boldsymbol{\eta}^{(a_l,a_e)}$ is a period tension and stays feasible, if $\boldsymbol{\eta}^{(a_l,a_e)} \leq \boldsymbol{\delta}$. Note, that the objective changes by

$$\boldsymbol{\omega}^t\mathbf{y}' = \boldsymbol{\omega}^t\mathbf{y} + \boldsymbol{\omega}^t\boldsymbol{\eta}^{(a_l,a_e)} - T\boldsymbol{\omega}^t\mathbf{z}'$$

Modulo Network Simplex Method

Initialisation: Determine an initial feasible tree structure $(\mathcal{T}^{\boldsymbol{\ell}}, \mathcal{T}^{\mathbf{u}})$ with feasible solution x

Iteration: **while** there exists an improving cut $\boldsymbol{\eta}$ **do**

1. apply this cut by transforming the solution $\mathbf{x} \rightarrow \mathbf{x} + \boldsymbol{\eta}$
2. solve a minumum cost flow problem to transform the actual solution to become a tree solution

3 Computational Results for a Real World Scenario

3.1 The Traffic Sample

We applied the described algorithm to a real world traffic sample, which was derived from the south-west area of the German Railway Network (see Figure 1).

The timetable problem contains 92 different railway lines with periods of 20, 30, 60 and 120 minutes, which results in an overall period of $T = lcm(20, 30, 60, 120) = 120$ minutes. The problem size of the resulting periodic event scheduling problem contains 669 event nodes and in total 3831 (with 3287 headway) constraints.

The feasibility problem without any passenger connection constraints, we used a constraint programming approach, which finds a feasible solution within approximately one minute computation time.

Next, for an origin destination matrix we applied a traffic assignment, by routing passengers on best paths. In this way we obtained for

each possible connection between different lines a weight for the number of passengers using this change activity. The origin destination matrix contains only values given in percent of the total (unknown) traffic volume. The results are given by table 1. For this reason, the change activity weights is primary that percentage of total volume which uses this connection. Due to the huge amount of approximately 1200 change activities with positive passenger weight, we only the most important ones, which are 570.

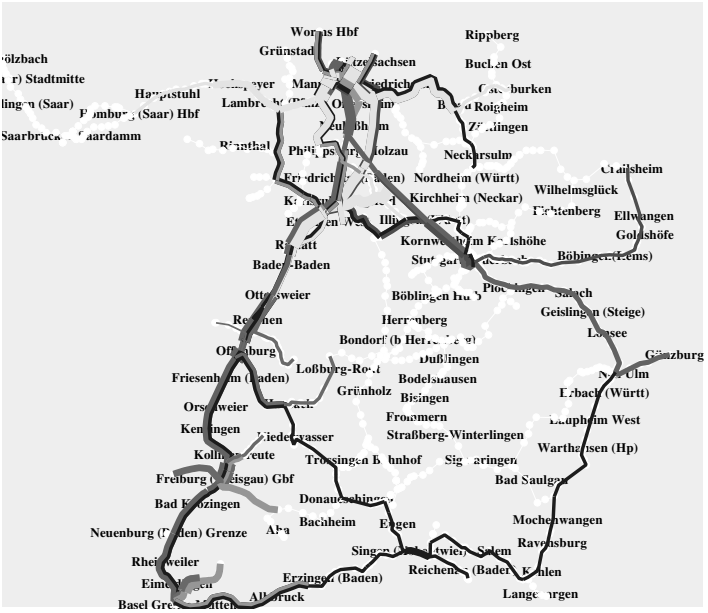


Fig. 1. The Traffic Sample contains 92 from the south-west area of the German railway network

4 Acknowledgment

The project was supported by the German Railway Company DB Netz AG.

Table 1. Computational results for the Modulo-Simplex-Algorithm

iteration	objective	description
	620952.00	initial solution from constraint propagation
	462111.00	min cost flow with fixed modulo parameter **z**
1	436881.00	modulo-netwok simplex
2	415182.00	modulo-network simplex
...	....	.....
68	254711.00	final solution

References

1. Kolonko, M. and Nachtigall, K. and Voget, S. (1996) Exponat der Universität Hildesheim auf der CeBit 96: Optimierung von integralen Taktfahrplänen mit genetischen Algorithmen Hildesheimer Informatik-Berichte 8/96
2. Liebchen, C. (2006) Periodic Timetable Optimization in Public Transport Dissertation, Institut für Mathematik, TU Berlin
3. Lindner, T. (2000) Train Shedule Optimization in Public Rail Transport Dissertation, TU Braunschweig
4. Nachtigall, K. (1996) Periodic Network Optimization with different Arc Frequencies Discrete Applied Mathematics 69:1–17
5. Nachtigall, K. (1998) Periodic Network Optimization and Fixed Interval Timetables Habilitationsschrift, Universität Hildesheim Institutsbericht (IB) 112-99/02, Deutsches Institut für Luft- und Raumfahrt, Braunschweig
6. Odijk, M.(1994) Construction of Periodic Timetables - Part I: A Cutting Plane Algorithm Technical Report, Department of Mathematics and Computer Science, University of Technology. Delft, The Netherlands
7. Peeters, L.W.P. (2003) Cyclic Railway Timetable Optimization Dissertation, Erasmus Research Institut of Management, Erasmus University, Rotterdam
8. Schrijver, A. (1986) Theory of Linear and Integer Programming, J. Wiley and Sons, Chichester New York Brisbane Toronto Singapore
9. Schrijver, A. and Steenbeek, A. (1994) Dienstregelingontwikkeling voor Railned Technical Report, Centrum voor Wiskunde en Informatica
10. Serafini, P. and Ukovich, W. (1989) A Mathematical Model for Periodic Scheduling Problems SIAM J. Discrete Math 4/2:550–581
11. Weigand, W. (1983) The Man-Maschine Dialogue and Timetable Planning Rail International 3:8–25

Simultaneous Vehicle and Crew Scheduling with Trip Shifting

András Kéri* and Knut Haase

Chair of Business Administration, Transport and Logistik, TU-Dresden
andras.keri@mailbox.tu-dresden.de, knut.haase@tu-dresden.de

Summary. Recent studies consider trip shifting as a possible way to include timetabling partially into vehicle scheduling for urban mass transit system planning. The aim of our research is to find an usable model to integrate trip shifting into Simultaneous Vehicle and Crew Scheduling (VCSP). The problem is solved usually by column generation, so trip shifting has to be modeled both at master problem and subproblem level. We present in detail an extension of the Resource Constrained Shortest Path Problem, which allows us to model the subproblem of the VCSP-TS problem.

Key words: Transportation and Logistics, Large Scale Optimization, Scheduling

1 Introduction

Sequential approches in the planning of an urban or sub-urban bus transportation system are outdated, because the global solution could be of very bad quality even it is easy to get an optimal solution for each step.

There are two important directions of the research for simultaneous approaches. One of them is based on the Simultaneous Vehicle and Crew Scheduling Problem (VCSP). The other one uses the trip shifting technique and tries to partially incorporate the timetabling into the vehicle scheduling problems (VSP-TS).

The aim of our research is to find a model, which combines trip shifting with the VCSP problem, and which can be solved within acceptable time on real-life instances.

* supported by DFG resarch grant 2843/3-1

2 The Model

Our approach is constrained to single depot, and homogeneous fleet problems. However, the model can be easily extended to multi-depot and heterogenous fleet.

It is a multicommodity network flow model, similar to that described in [3]. The column generation technique is well suited to solve it, so we have divided the model into a main and a subproblem.

The main problem is a generalized set covering problem with side constraints, where the variables are the feasible paths. The subproblem - which is a multicommodity-flow network with resource constraint on arcs - is responsible for generating the feasible paths. It can be solved as a Resource Constrained Shortest Path Problem (RCSPP).One can use a branch-and-cut algorithm or a round-up heuristic to obtain an integer solution.

For modelling trip shifting, we assign to each trip a set of possible shifting times, which is a relative value to the original starting time resulted from the timetabling step.

2.1 The Main Problem

Let W be the set of trips, indexed by w. Each trip is divided into tasks (d-trip), which have to be performed by drivers. Let V represent the set of tasks, indexed by v. The set of possible shifting times of trip w is denoted by S_w. Let H represent the set of the depot leaving times, indexed by h. Let U be the set of duty-types, indexed by u. The set of feasible duties are $\Omega^u, u \in U$, indexed by ρ.

The following binary parameters are used in the model: d_v^ρ is 1, if task v is covered by duty ρ, otherwise 0. e_w^ρ is 1, if duty ρ contains a driving movement which ends at the start station of trip w, otherwise 0. f_w^ρ is 1, if duty ρ contains a driving movement which starts at the end station of trip w. Let q_h^ρ equal to one, if duty ρ contains a driving movement starting before time point $h \in H$, and ending after h, otherwise 0.

The binary variable $\theta_\rho^u, \rho \in \Omega^u, u \in U$ represents whether duty ρ is performed or not. The integer variable B is the minimal number of buses required to cover the schedule. The binary variable $X_{w,t}$ is a time-indexed variable, it is equal to one if the trip w is shifted by t time units ($t \in S_w$).

The model of the combined vehicle and crew scheduling with trip shifting is the following:

$$\min c_B B + \sum_{u \in U} \sum_{\rho \in \Omega^u} c_\rho \theta_\rho^u \tag{1}$$

$$\sum_{u \in U} \sum_{\rho \in \Omega^u} d_v^\rho \theta_\rho^u = 1, \qquad \forall v \in V, \tag{2}$$

$$\sum_{u \in U} \sum_{\rho \in \Omega^u} e_w^\rho \theta_\rho^u = 1, \qquad \forall w \in W, \tag{3}$$

$$\sum_{u \in U} \sum_{\rho \in \Omega^u} f_w^\rho \theta_\rho^u = 1, \qquad \forall w \in W \tag{4}$$

$$\sum_{u \in U} \sum_{\rho \in \Omega^u} q_h^\rho \theta_\rho^u \leq B, \qquad \forall h \in H \tag{5}$$

$$\sum_{t \in S_w} X_{w,t} = 1, \qquad \forall w \in W \tag{6}$$

$$X_{w,t}, \theta_\rho^u \text{ are binary}, \quad B \text{ is integer.} \tag{7}$$

The objective function (1) minimizes first the number of buses (here c_B is a large enough number), then the crew cost. The equations (2) represent the task-covering constraints. Each task v has to be covered exactly once. (3) and (4) are the so-called bus flow conservation equations. There should be exactly one driving movement ending at the start station of trip w, and one driving movement starting at the end station of trip w. (5) are the bus counting inequalities. The equations (6) force that for each trip will be exactly one shifting time assigned.

The solution contains only the duties that have to be performed by the crew. However, with a polynomial time algorithm one can generate the vehicle blocks from this solution.

2.2 The Subproblem

The sub-problems are Resources Constrained Shortes Path Problems, and the underlying networks are time-space networks. Each node in the network has a fixed time point and a fixed location. The arcs represent driver movements, which can be a walking movement, or a driving movement. The labour regulations are modeled by resources. For each type of duties (normal duty, night duty, trippers, etc.) exists different labour regulations, thus we have as many sub-problem as the number of duty types. A more detailed description can be found in [3].

For modeling trip shifting, we use the same approach as described in [1]. Namely, each task is represented by as many task-covering arcs as the number of possible shifting times of the corresponding trip. Figure 1

illustrates this. There are two trips on it, both of them having three possible shifting time $\{-2, 0, 2\}$.

The reduced cost of the task-covering arcs can be calculated from equalities (2) and (6).

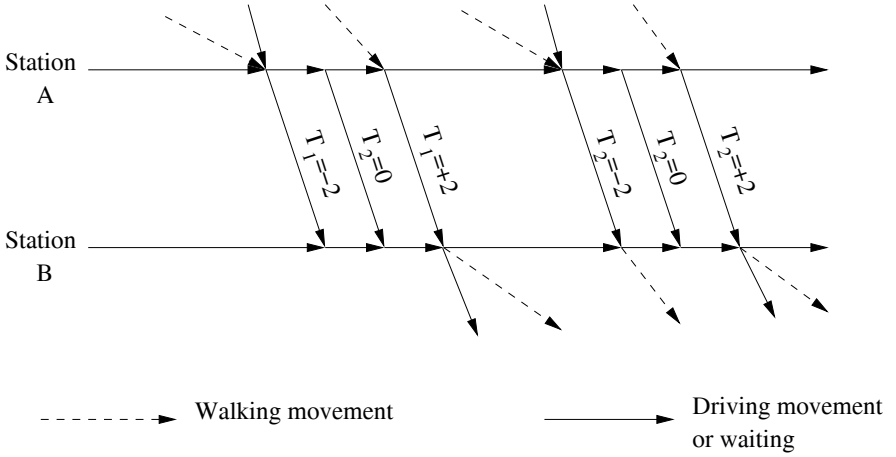


Fig. 1. Modelling trip shifting in the sub-problem

For solving the subproblem, we are using a labelling algorithm described in [2]. To accelarate the method, we do not explore all of the labels in a node, just the n-most-negative-reduced-cost ones. Exploring all label is only neccessary for prooving the optimality in the final steps.

3 Extensions of the Model

3.1 Precedence Constraint

If for some reason the connections between trips are important, we can model this by adding precedence constraints on $X_{w,t}$ variables to the model. With this construction, we can reduce the waiting time of the passangers.

Suppose that there are two trips w_1 and w_2, where the end-station of w_1 is the same as the start-station of w_2, and the ending time of w_1 is larger than w_2 by one minute. This prevents passengers to change from trip w_1 to trip w_2. If we would like to allow one minute changing time for the passengers, we have to force that the end time of trip w_1 will be smaller than the start time of w_2 by one minute. It means that

the shifting time of w_1 has to be smaller at least by two minutes than the shifting times of w_2, since the shifting times are relative times. We can model this with the following precedence relation:

$$X_{w_2,t_2} + \sum_{\substack{t \in S_{w_1} \\ t > t_2 - 2}} X_{w_1,t} \leq 1 \qquad \forall t_2 \in S_{w_2} \tag{8}$$

In the subproblem, these constraint can be modeled by generating only those kind of intertrip arcs, that fulfills the required precedence relations.

3.2 Flexible Groups

An interesting extension of the model is when one partitions the set of trips into distinct subsets, and the trips in the same subset have the same shifting time. We call these subsets *flexible groups*.

Let $g \in G$ be the set of flexible groups. W_g is the set of trips in the flexible group $g \in G$. The group corresponding to trip w is denoted by g_w. The set of possible shifting times of the trips in group g is S_g. The parameter $k^{\rho}_{g,t}$ is 1, if duty ρ fixes the shifting time of group g at time point t. The binary variable $Y_{g,t}$ is a time-indexed variable, it is equal to one if the flexible group g is shifted by t time units ($t \in S_g$).

To using flexible groups, one has to replace (6) with (9)-(10) in the main problem.

$$\sum_{u \in U} \sum_{\rho \in \Omega^u} k^{\rho}_{g,t} \theta^u_{\rho} \leq c_L Y_{g,t} \qquad \forall g \in G, \forall t \in S_g \tag{9}$$

$$\sum_{t \in S_g} Y_{g,t} = 1, \qquad \forall g \in G \tag{10}$$

(9) are compatibility contraints, namely, they ensure that a solution contains only such a paths, for which the same group has the same shifting time. The equations (10) force that each group has exactly one starting time.

In the subproblem, since more trips share the same shifting time, we have to keep track of the selected shifting times for all group. We store the shifting time of each group in the labels. Since the groups contains more than one trips, it is possible that more task covering arcs corresponding to different trips represent the same group and shifting time combination. To avoid the situation, that a path selects different

shifting times for the same group, it has to be always checked on a task-covering arc, whether the corresponding group has already a shifting time, or not. If it is so, the given arc can only be used, if the shifting time is equal with the previously selected value.

The dominance rule also has to be slightly modified. Two labels in a node are only compatible (the domince rule can be applied on them) if the already selected group shifting times are equal for both of them.

The advantage of flexible groups is that it can be used to avoid one of the side effects of trip shifting, the non-constant time headways, and can reduce the size of the main problem.

It is also possible to define precedence constraints between two flexible groups. This makes sense, if the headway between the trips in both groups are equal.

4 Conclusions

In this article we have presented a new model for the VCSP-TS. The model is based on a column generation approach. We proposed two possible extensions: i) the addition of precedence constraints between trips, that can be used to reduce the waiting time of changing passangers, and ii) the usage of flexible groups, which can help to avoid non-constant time headways and it can reduce the size of the problem.

References

1. Bunte, S., N. Kliewer, L. Suhl (2006) Mehrdepot-Umlaufplanung: Berücksichtigung von Verschiebeintervallen für Fahrten in einem Time-Space-Netzwerk-basierten Modell, In: Haasis H., Kopfer H., Schönberger J.: Operations Research Proceedings 2005 - Selected Papers of the Annual International Conference of the German Operations Research Society (GOR), University of Bremen, Germany, Springer
2. Desrochers, M. (1986) La fabrication d'horaries de travail pour les conductors des autobus par une méthode de génération de colonnes, Ph. D. Dissertation, Université de Montréal, Montréal, In French
3. Haase, K., G. Desaulniers, J. Desrosiers, (2001) Simultaneous Vehicle and Crew Scheduling in Urban Mass Transit Systems, *Transportation Science* 35(3):286–303

Line Optimization in Public Transport Systems

Michael J. Klier and Knut Haase

Chair of Business Administration, Transport and Logistics, TU Dresden
michael.klier@tu-dresden.de, knut.haase@tu-dresden.de

1 Introduction

Line planning is one of the strategic tasks a transport company is faced with. The aim is to create a line plan with line routes and service frequencies. Line optimization means to determine a line plan that is optimal regarding to a defined objective like the number of direct travelers [3], the total ride time, number of changes [5], the total cost [2] or the total traveling time. The literature offers approaches with choosing lines from a given set as well as construct line routes from the scratch [1], [4].

All of these approaches presume a given origin-destination-matrix. At least for urban areas this is not realistic. The most important questions of a traffic planner of a transportation company are: "How much does the new line plan cost?" and "How many passengers will go by public transport under the new circumstances?". Obviously it is necessary to consider the movement in demand for public transport within line optimization.

In this paper we include frequency depending changing times. In urban public transport systems often more than one line connects two points in a direct way. The expected traveling time is therefore lower than riding time plus half of the frequency time of the used line(s). The waiting times will decrease if there are e.g. two lines that connect two points by parallel line routes.

In practice, transport companies take advantage of lines that are parallel in the city and separate in the periphery to give a good service in the area with a great demand and connect the suburbs more efficient with the city. By experience (i.e. tested with data of Dresden)

minimizing traveling times without regarding parallel line routes yields unrealistic results for the waiting times.

2 Model

In this section we present a model that can cope with (partially) parallel lines and traveling time dependent passenger demand.

2.1 Assumptions

Let $G[V, A]$ be a directed graph with a set of nodes V and a set of node connecting arcs A. The nodes represent stops for public transport. The arcs symbolize connections between nodes that can be passed by public transport vehicles. For each arc a ride time t_{ij} is defined. Furthermore, we know a set of line routes L. The arcs (i, j) which are part of the route of line l are given by set $\hat{A}_{lij}$. F is a set of possible frequencies a line can be operated with.

Every node pair that is connected by at least one potential line route yields for each combination of potential line routes l and frequencies f one arc. The larger the pool of lines L is, the more such arcs are required. For practical reasons we generate combinations with no more than five parallel line routes. Thus for each combination the expected traveling time can be estimated. Furthermore, we are able to calculate the proportion of passengers for each line frequency combination within a subset of lines. We assume that a path $p \in P$ is a connection between one pair of nodes u, v i.e. possibility for passengers to get from node u to node v. While the arcs are direct connections by one or more lines a path can be a combination of more than one arc. So necessary changes on the way from u to v can be modeled.

Example We show the computation of traveling times for one path. In Figure 1 you can see the connection between node u and node v with three lines. Line 2 connects the origin and the destination directly with a detour of four minutes and a frequency of 2 vehicles per hour while line 1 and line 3 offer only a part of this connection. We are now able to calculate traveling times (including the expected waiting times) of the given connections. Exemplary the traveling times of two generated arcs are shown in Figure 2.

The traveling times of the arcs are calculated as follows:

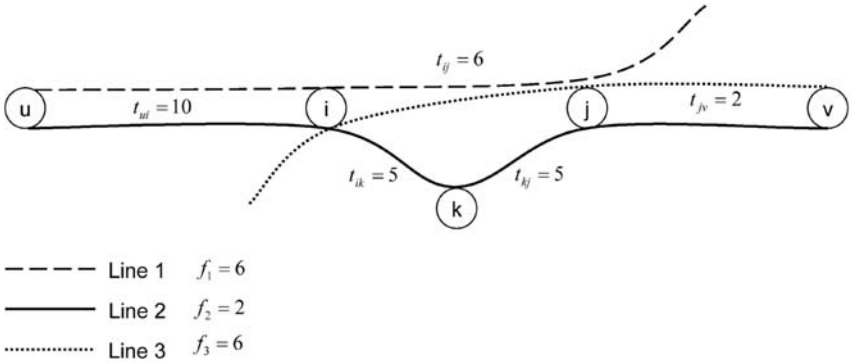


Fig. 1. Generated path with three involved lines

$$t^*_{pui} = \underbrace{\frac{60}{2 \cdot (2+6)}}_{\text{waiting time}} + \underbrace{\frac{2 \cdot (10+10) + 6 \cdot (10+6)}{(2+6)}}_{\text{riding time}} = 20.75 \tag{1}$$

$$t^*_{piv} = \frac{60}{2 \cdot (2+6)} + \frac{2 \cdot 2 + 6 \cdot 2}{(2+6)} = 5.75 \tag{2}$$

The expected traveling time of the shown path is $t^*_{puv} = 20.75 + 5.75 = 26.5$ minutes. If only line 2 is available for the path, the expected traveling time is $15 + 22 = 37$ minutes. Similar to this we can calculate the proportions of the demand of each original arc and each line frequency combination as follows:

$$\beta_{p,1,6,i,j} = \frac{6}{6+2} = 0.75 \tag{3}$$

This means that 75% of the demand of passengers from u terminating in v will go by line 1 with the frequency of 6 vehicles per hour on arc (i, j) when the line plan contains path p. We assume that 75% of the

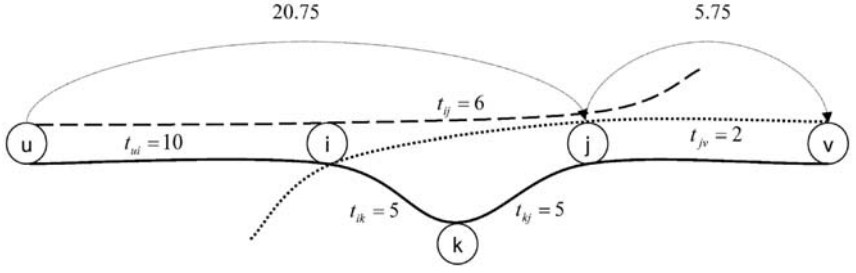


Fig. 2. Traveling times under the condition of parallel line routes

passengers will go by line one because six out of eight vehicles per hour belong to line 1.

On the basis of generating a large set of paths before optimization (e.g. by a n-shortest-path-algorithm) we get the following model:

$$\max F = \sum_{p} d_p \cdot z_p \tag{4}$$

$$\sum_{p,u,v \in \bar{p}_{puv}} z_p \leq 1 \qquad \forall\,(u,v) \in V^2 | u \neq v \tag{5}$$

$$\sum_{f} y_{lf} \leq 1 \qquad \forall l \in L \tag{6}$$

$$\sum_{p \in \hat{P}_{plfij}} d_p \cdot \beta_{plfij} \cdot z_p \leq K_{lf} \cdot y_{lf} \qquad \forall\,(l,i,j) \in \hat{A}_{lij}, \forall f \in F \tag{7}$$

$$\sum_{l,f} c_{lf} \cdot y_{lf} \leq C \tag{8}$$

$$z_p \in \{0,1\} \qquad \forall p \in P \tag{9}$$

$$y_{lf} \in \{0,1\} \qquad \forall l \in L, \forall f \in F \tag{10}$$

The objective function (4) maximizes the expected total number of passengers. To every path p an expected traveling time is assigned, that defines the expected number of passengers d_p. The binary variables z_p decide whether path p is selected or not. Obviously one pair (u,v) of nodes can be connected by many different paths. The constraints (5) ensure that the demand of one node pair (u,v) can be met by maximum one path p. The set $\bar{p}_{puv}$ gives for each path p the corresponding origin and destination. It is allowed to choose at most one frequency f for every line path l (6). The binary variable y_{lf} take the value 1, when line route l with the frequency f is selected. The capacity constraints (7) for all arcs (i,j), belonging to the line route l and the frequency f, give at least the capacity to handle the number of passengers moving along it. The demand of a path d_p multiplied by β_{plfij} represents the expected number of passengers. Set $\hat{P}_{plfij}$ denotes line routes l frequencies f and arcs (i,j) which correspond to path p. The parameter K_{llf} gives the capacity per vehicle of line l and frequency f. Let c_{lf} be the (proportional to the riding time) operating cost of line route l and frequency f. So the constraint (8) bounds the total operating cost to a given maximum of total cost C.

2.2 Discussion

One obvious problem is the large amount of possible and reasonable paths. When generating them before the optimization process we enlarge the model unnecessarily because most of the paths will not be part of a solution. So it seems to be appropriate to generate only those paths which will be probably part of a solution. A decomposition method for problems with many possible but only a few reasonable alternatives could be helpful.

3 Example

To clarify the above statements we present a small example. Starting with a directed graph with 10 nodes and 36 arcs. Before the optimization process we defined 16 possible line routes. Based on it 4320 arcs have to be created to model all (parallel) line frequency combinations. After that 24728 possible paths are generated by a modified n-shortest-path-algorithm. These paths contain the line routes, the frequencies and the passenger demand for each arc of the original graph.

For each pair of nodes we assume linear demand functions $d_{puv} = a_{uv} - b_{uv} \cdot t^*_{puv}$ with random parameters a_{uv} and b_{uv}.

Now we are in a position to solve the problem and vary the maximum total cost. The results are shown in table 1. One obvious result is that the increase of the maximum total cost yields no fundamental increase of the total number of expected passengers at a certain point.

Table 1. Results of the example

No	max. Cost	objective	computing time [s]	gap
1	100	1016	1.92	-
2	200	1852	33.04	-
3	400	2840	1000.00	0.038
4	500	3228	1000.00	0.033
5	700	3621	31.28	-
6	800	3631	14.26	-
7	1600	3656	7.62	-

For all scenarios, which took more than 1000 seconds the gap between the best possible and the actual integer solution is denoted. For example in Figure 3 we show the solution of scenario 5 with the maximum total cost of 700 minutes of total vehicle operating time and the objective of 3621 passengers.

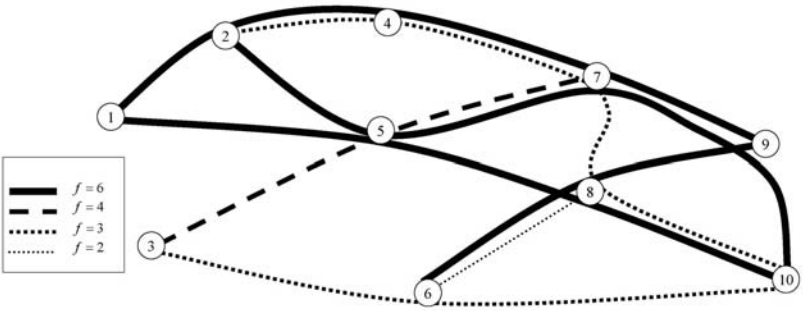


Fig. 3. Solution of scenario 5

4 Conclusions

In this article we have presented an approach on line optimization in urban public transport systems. It is shown that it is possible to take into account parallel line routes (with decreasing waiting times) and changing demand. There is still a lot of work to be done in the field of estimation of relation specific demand regarding to the expected traveling time and e.g. the number of changes needed or socioeconomic structures of corresponding districts. Moreover, the solution process should be made more efficient to get the ability for solving real world instances. An advantage of our approach is that it is possible to include non-linear demand functions in the data but nevertheless the model will stay linear.

References

1. R. Borndörfer, M. Grötschel, and M.E. Pfetsch. Models for line planning in public transport. *ZIB-Report, Konrad-Zuse-Institut Berlin*, 04-10, 2004.
2. M. R. Bussieck. *Optimal lines in public transport.* PhD thesis, Technische Universität Braunschweig, 1997.
3. H. Dienst. *Linienplanung im spurgeführten Personenverkehr mit Hilfe eines heuristischen Verfahrens.* PhD thesis, Technische Universität Braunschweig, 1978.
4. M.E. Pfetsch and R. Borndörfer. Routing in line planning for public transport. *ZIB-Report*, 05-36, 2005.
5. S. Scholl. *Customer-Oriented Line Planning.* PhD thesis, Technische Universität Kaiserslautern, 2005.

Coordination in Recycling Networks

Eberhard Schmid, Grit Walther, and Thomas S. Spengler

Technische Universität Braunschweig, Lehrstuhl für Produktion und Logistik, Katharinenstr. 3, 38106 Braunschweig, Germany
{e.schmid|g.walther|t.spengler}@tu-bs.de

Summary. We consider a legislation-driven recycling network treating discarded products. Our goal is to develop a decentralised coordination mechanism that allows the network to comply with requirements given by environmental legislation, existing for Waste Electric and Electronic Equipment (WEEE). Two decision levels are identified. Tactical decisions concern the negotiation of frame contracts between a focal company representing the network, and the recycling companies for a defined period of time. According to these frame contracts, current orders are assigned to recycling companies and an operational coordination may be applied to (re-)allocate parts of the order in the network. The mechanisms are outlined conceptionally and their interaction is described.

1 Introduction

Legal regulations in the field of WEEE recycling assign extended product responsibility to Original Equipment Manufacturers (OEMs). Hence, OEMs have to pay for the recycling of their own products and they are obliged to guarantee the fulfilment of certain collection and recycling targets. However, since recycling normally falls outside the core competence of OEMs, these tasks are usually transferred to third-party recycling companies. Empirical studies show that WEEE recycling companies are often small and medium sized companies, operating locally and cooperating in networks [3].

One focal company and multiple recycling companies exist within these networks. The focal company is often founded by the network members, and therefore aims at maximising network-wide profit. It's job is to acquire national or Europe-wide recycling contracts with OEMs specifying the collection and recycling of a certain amount of

discarded products accumulating at certain collection points for a defined period of time. Subsequently, the focal company has to make sure that these contracts are adequately fulfilled. The physical treatment of the discarded products is performed by independent recycling companies. They pick up discarded products at the sources of the network which are public collection points. Products are then transported to recycling companies where disassembly and bulk recycling are carried out. Thereby, the grade of disassembly and bulk recycling may influence the obtained recycling targets as well as the recycling costs. Thereafter, generated material fractions are either delivered to sinks of the network, which are either recovery or disposal facilities (e.g. metal works, incineration) or to other recycling companies in the network which perform an advanced treatment of the fractions in order to reach higher recycling targets. To determine recycling targets, we assume that a fraction- and sink-specific recycling coefficient can be assigned to every material fraction leaving the network. The companies have to agree upon the material flows by generating appropriate contracts. Contractual agreements and material flows are summarised in Figure 1.

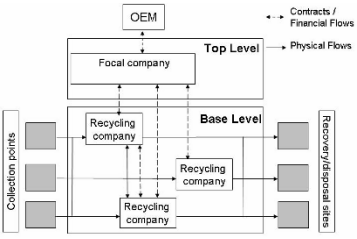


Fig. 1. Material flows and contractual agreements in recycling networks

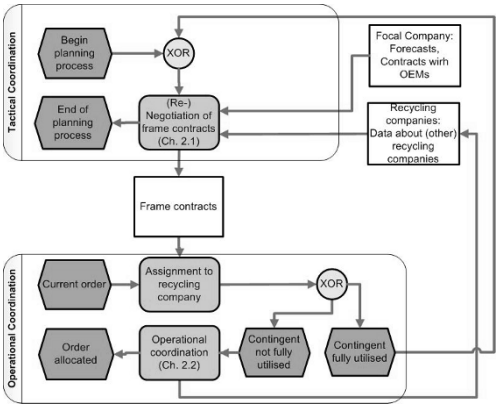


Fig. 2. Overview of the entire coordination process

The goal of this paper is to highlight the need for coordination in such a network and to identify coordination levels. The coordination levels and their interaction are described in the following section (see also Figure 2).

2 Coordination Levels in Recycling Networks

Considering the situation described above, extended need for inter-company coordination becomes apparent, since decisions are made by independent decision makers. First, the focal company and the recycling companies have to agree upon the allocation of the masses accumulating at the sources to the recycling companies. Second, after the recycling companies have decided on the recycling grade, they have to agree with each other whether either a specialised processing of fractions within the network is necessary (which would result in flows between the recycling companies) or if fractions are directly delivered to the sinks of the network. Coordination should be carried out in an efficient way taking into account global restrictions (recycling and collection targets) and respecting the decision authority of the independent companies.

If full information was available at any point in time and one partner – e.g. the focal company – had the power to implement network-wide plans, central optimisation approaches could be implemented [4]. Since this case rarely exists in a network consisting of independent companies, a decentralised coordination approach is required. As decisions are not generally taken at the same time and require different degrees of aggregation, a hierarchical decomposition of planning modules is reasonable [1]. As with classical hierarchical planning within a single firm, it is reasonable to decompose the coordination of a network into different levels. On a *tactical level*, the focal company negotiates frame contracts with the recycling companies picking up the products from the sources and performing the first treatment step. These contracts contain specifications on the masses that each company has to pick up in specified period of time (e.g. a year) and recycling targets to fulfil individually. Such contracts also exist in real world recycling networks. On an *operational level*, current orders that have been assigned by the focal company according to frame contracts eventually have to be exchanged or divided up within the network. In the following the levels are described in more detail.

2.1 Tactical Coordination

In the tactical coordination phase the focal company negotiates with the recycling companies about the masses to be picked up from the sources and corresponding recycling targets to be fulfilled within a defined period of time. Further, associated payments are determined. In

this level, simultaneous coordination of flows between recycling companies is not considered. Each recycling company considers the other recycling companies as sinks ("dummy sinks"). To determine recycling targets, we assume the each recycling company has enough knowledge to anticipate fraction-specific recycling coefficients for fractions delivered to other recycling companies inside the network and the respective prices, as it has for the real sinks (see Section 1). Based on these simplifications, an independent optimisation model for each recycling company can be established. The prices paid by the focal company for the masses picked up from sources and the prices paid for each recycled mass unit influence the objective function of each recycling company. These prices result from the (Lagrangian) relaxation of network-wide recycling and collection constraints and are iteratively updated until a feasible solution is found. This procedure can be interpreted as a hierarchical negotiation procedure where the focal company acts as the Top-Level and the recycling companies as the Base-Level (see again Figure 1). Details and numerical evaluation of this model can be found in [2]. These frame contracts are valid for specified periods of time and can be updated on a rolling basis (see Figure 2).

2.2 Operational Coordination

Current orders for the collection of WEEE are collected by a central agency of the WEEE management system and assigned to the service provider (=focal company, representing an OEM) according to a predefined scheme. As shown in Figure 2, the focal company can then assign the order to a recycling company according to the frame contracts negotiated, as described in Section 2.1 (If the negotiated contingent is fully utilised, a renegotiation of contracts may be necessary). The recycling company to which the order has been assigned, may need services from other recycling companies in the network in order to reach the specified recycling targets. This means that parts of the order may have to be allocated to other recycling companies. To this end, we propose the following simple rule-based mechanism to achieve coordination on the operational level. In the following, we assume that the focal company has assigned recycling company u an order to collect masses of different product types i at different collection points q ($\overline{A_{iqu}}$) and a certain amount of mass that has to be recycled ($\overline{y_u^{Rec}}$). According to the frame contract the focal company pays p_{iq} per mass unit collected and λ per mass unit recycled [2]. The mechanism can be outlined as follows.

1. **recycling company** u:
   - F denotes the set of fractions (including products), Q the set of sources, R the set of sinks outside the network and D the "dummy sinks" within the network (other recycling companies). y^R_{iur} denotes the mass of fraction i which is delivered to sink (resp. dummy sink) r and e^V_{ir} the associated costs/revenues. x_{ju} state the number of executions of recycling activity j and c_{ju} are the associated costs. The recycling company u then solves the following problem:

$$\sum_{i\in F}\sum_{q\in Q} p_{iq}\cdot\overline{A_{iqu}} + \lambda\cdot\overline{y^{Rec}_u} + \max \sum_{r\in R\cup D}\sum_{i\in F} e^V_{ir}\cdot y^R_{iur} - \sum_{j\in J} c_{ju}\cdot x_{ju} \qquad (1)$$

   s.t.

$$\sum_{q\in Q}\overline{A_{iqu}} + \sum_{j\in J} x_{ju}\cdot v_{ij} = \sum_{r\in R\cup D} y^R_{iur} \qquad \forall i\in F \qquad (2)$$

$$\overline{y^{Rec}_u} \le \sum_{i\in F}\sum_{r\in R\cup D} y^R_{iur}\cdot\chi_{ir} \qquad (3)$$

   The objective function (1) consists of the fixed payments from the focal company and variable costs/revenues resulting from material fractions delivered to sinks or to other recycling companies as well as recycling costs. (2) is the mass balance equation. v_{ij} is the recycling coefficient, which represents the mass units generated (+)/consumed(-) from each fraction i by each execution of recycling activity j. (3) is the recycling target that has to be fulfilled for the current order. χ_{ir} denotes the recycling coefficient which denotes to what extent fraction i is recycled when delivered to sink/dummy sink r.
   - The assignment of material fractions to the dummy sinks is tentative and based on presumed data. Since companies in the network, their recycling costs, or their abilities to process certain types of fractions may have changed, a request for recycling of fractions that have been assigned to dummy sinks is sent to all other recycling companies in the following way:
     Choose fractions that have been assigned to the dummy sinks ($r \in D$) and compute assigned recycled masses for each fraction
   - For each fraction submit a bundle containing mass and assigned recycled mass to all other recycling companies in the network.
2. **all other recycling companies**:
   - Compute cost/revenue that is caused by treating the additional bundle(s) of fractions with the required recycling target.
   - Submit associated cost/revenue back to recycling company u

3. **recycling company** u:
   - Choose cheapest recycling company.
   - Agree with recycling company upon a price.
   - Update prices for the dummy sinks (e_{ir}^{V}) that can be used for the tactical negotiation of frame contracts (see Figure 2).

The operational coordination procedure is still subject to research. Open questions concern the exchange of cost data that is necessary for the proper operability of the mechanism and the determination of prices to pay. Also not only parts of an order (generated material fractions for further processing), but also complete orders (pick up and first treatment step) may be exchanged between the recycling companies if the cost situation has changed significantly.

The whole coordination procedure containing tactical and operational coordination and their interaction is summarised in Figure 2.

3 Conclusions and Outlook

In this paper we outlined a negotiation mechanism for recycling networks operating in the highly legally restricted area of WEEE recycling. We identified two coordination levels that exist in such networks and described their interaction. Future research will concentrate on the implementation of the operational coordination procedure.

References

1. B. Fleischmann and H. Meyr. Hierarchy, modeling and advanced planning systems. In S. Graves and A. De Kok, editors, *Handbooks in Operations Research and Management Science: Supply Chain Management*, pages 457–523. Elsevier, Amsterdam, 2003.
2. G. Walther, E. Schmid, and T. Spengler. Negotiation based coordination in product recovery networks. *International Journal of Production Economics*, 2007. Forthcoming.
3. G. Walther and T. Spengler. Empirical analysis of collaboration potentials of sme in product recovery. *Progress in Industrial Ecology - An International Journal*, 1(4):363–384, 2004.
4. G. Walther and T. Spengler. Impact of WEEE-directive on reverse logistics in Germany. *International Journal of Physical Distribution and Logistics Management*, 35(5):337–361, 2005.

Produktsegmentierung mit Fuzzy–Logik zur Bestimmung der Parameter eines Lagerhaltungsmodells für die Halbleiterindustrie

Alexander Schömig[1] and Florian Schlote[2]

[1] Infineon Technologies AG, AIM SCM BS SCI, 81726 München
alexander.schoemig@infineon.com
[2] Infineon Technologies North America Corp., San Jose, CA, U.S.A.
florian.schlote@infineon.com

1 Einleitung

Die Implementierung einer geeigneten Bevorratungssystematik ist eine Aufgabe, die von jedem produzierenden Unternehmen gelöst werden muss. In der Literatur existieren vielfältige Lagerhaltungsmodelle für unterschiedliche Anwendungszwecke, ebenso gibt es eine große Anzahl kommerziell verfügbarer Softwarepakete, die diese Planungsaufgabe unterstützen sollen. Sowohl die grundlegende Entscheidung, für welche Produkte überhaupt ein entsprechender kundenauftragsanonymer Lagerbestand vorgehalten werden muss, die Bestimmung von Anzahl und Ort von Lagern, als auch die Parametrisierung des gewählten Lagerhaltungsmodells, erfordern eine Analyse der entsprechenden Rahmenbedingungen. Diese bleibt in der Regel dem Anwender überlassen.

In den Jahren 2002 – 2004 wurde bei der Infineon Technologies AG, München, eine praktikable Systematik zur Festlegung der Produktbevorratungsebene entwickelt. Es entstand ein proprietäres Entscheidungsunterstützungssystem mit dessen Hilfe die Festlegung der Bevorratungsebene für das Produktionssortiment durchgeführt wird. Die Kernidee dieses Modells ist die Segmentierung der Produkte hinsichtlich ihrer Bevorratungsebene.

Klassischerweise erfolgt dies mit Hilfe einer ABC-Klassifikation, oft auch unter Hinzuziehung eines zusätzlichen Kriteriums (zur Kritik einer eindimensionalen ABC-Betrachtung siehe [1].) Dies ist auch in der Halbleiterindustrie die typische Vorgehensweise.

2 Fertigungs- und Produktstruktur der Infineon Technologies AG

Hinsichtlich der Fertigungsstruktur werden üblicherweise zwei Fertigungssegmente unterschieden: Das *Frontend* und das *Backend*. Die Details dieser Struktur, sowie der Produktion einem Netzwerk von Fertigungsstandorten werden in [2] ausführlich diskutiert. Frontend und Backend werden durch die sogenannte *Diebank* entkoppelt. Sie stellt ein klassisches Zwischenlager dar, für das Backend ein Eingangswarenlager. Fertigprodukte werden in den regionalen Verkaufslagern (*Distribution Center*, DC) gelagert; standardmä"sig werden alle Aufträge über diese Lager bedient. Im Sinne einer kundenanonymen Vorfertigung gibt es weiterhin innerhalb des Frontends ein so genanntes *Masterlager* (Master Storage). Bis hierhin werden Wafer bis zu einem gewissen Grad angearbeitet, dann wird auf die entsprechende Kundenspezifikation der letzten Strukturschichten gewartet. Diese Bevorratungsstufe findet nur für einzelne Produkte Anwendung. Zur Identifikation von Produkten im Produktionsablauf existiert die *Baunummer*. Die DC-Baunummer stellt die kleinste Lageridentifikationseinheit (engl.: *stock-keeping unit*) dar. Viele planungsrelevante Attribute können nur auf dieser Endproduktebene vergeben werden.

3 Festlegung der Bevorratungsebene

Auch in der Halbleiterindustrie kommt den Kundenanforderungen an die Produktvielfalt und -verfügbarkeit eine verstärkte Bedeutung zu. Dies ist kritisch besonders im Hinblick auf die Einhaltung kurzer Lieferzeiten trotz kundenindividueller Variantenvielfalt. Dabei sind für ein Unternehmen des Weiteren Kosten- und Effizienzziele zu beachten, so dass zur Lösung dieses Konfliktes zusätzliche Anforderungen an die Logistik- und Supply-Chain-Management-Systeme der Unternehmen gestellt werden. Olhager [4] sieht zudem durch verringerte Produktlebenszyklen einen erhöhten Handlungsbedarf bei der Entscheidung über die zu wählende Bevorratungsebene.

Die Segmentierung des Produktsortiments ermöglicht eine systematische, standardisierte Planung der Lagerbestände. Ziel ist es, die Produkte zu identifizieren, für die eine kürzere Lieferzeit angeboten werden soll und diesen eine entsprechende Bevorratungsebene zuzuweisen. Diese Bevorratungsebene wird in diesem Kontext in der Literatur u. a. als *Order Penetration Point* (OPP) bezeichnet. Dieser sei definiert als derjenigen Punkt, an dem ein Halbfertig-Produkt einem bestimmten

Kunden zugeordnet wird (vgl. [5]). D.h. ab dem OPP ist der Materialfluss kundenauftragsgetrieben: Oberhalb des OPP erfolgt die Fertigungsauftragsfreigabe kundenanonym, unterhalb auftragsspezifisch. Die Wahl eines OPP steht dabei mit einem Merkmal zur Klassifikation von Produktionssystemen in enger Relation. Unter dem Ausdruck Segmentation soll das systematische Festlegen der Bevorratungsebene verstanden werden. Die Segmentation ist endproduktbezogen, und ist von der Fertigungssegmentierung abzugrenzen.

Im Rahmen der Segmentation wird bei Infineon ein so genannter *Service Level Code* vergeben. Dieser legt die letzte Lagerstufe fest, an dem für ein Produkt noch ein kundenanonymer Lagerbestand gehalten wird. Für diesen und die davor liegenden Lagerpunkte werden Sicherheitsbestände systematisch geplant. Der Service Level Code kann auch als Bezeichnung für den Order Penetration Point angesehen werden. Zur Auswahl stehen die oben vorgestellten Lagerstufen. Der Service Level Code wird in einem quartalsmäßigen Review festgelegt.

Der Erstentwurf des Segmentationsprozess bei Infineon fokussierte sich auf die Hauptkriterien Anzahl und Wichtigkeit der unterschiedlichen Endkunden eines Produktes und die Marktklassifikation. Obwohl weitere Kriterien in der Prozessbeschreibung erwähnt waren, konnten diese nicht automatisiert bei der Vorschlagserstellung berücksichtigt werden. Ein weiterer Kritikpunkt an diesem Prozess war hauptsächlich auf die Verknüpfung der Kriterien bezogen. Die harten Grenzen führen bei Nichterfüllen bereits eines Kriteriums zu einem Ausschluss einer Entscheidungsmöglichkeit. Diese Kritikpunkte haben eine erfolgreiche Umsetzung der erstellten Service Level Code Vorschläge erschwert und sind auf Akzeptanzprobleme bei den Logistikplanern gestoßen.

4 Die Wahl der Bevorratungsebene

Bei der Wahl eines optimalen Order Penetration Point sind teilweise gegenläufige oder schwer quantifizierbare Einflüsse zu beachten:

- Höhe des Obsoleszenzrisikos.
- Entstehende Lagerkosten durch die (Teil-) Vorfertigung.
- Akquisitorisches Potential durch verkürzte Lieferzeiten. (Durch das zielgenaue Erfüllen von Kundenlieferzeitanforderungen kann ein hohes Niveau an Kundenzufriedenheit erreicht werden.)
- Auswirkungen der Vorfertigung auf eine wirtschaftliche Produktion und die Arbeitsbedingungen.

4.1 Vorgehensweise

Bei der Wahl der Bevorratungsebene können durch Hinzuziehen vieler zu berücksichtigender Systemparameter leicht komplexe Entscheidungssituationen entstehen. Zur Unterstützung des Entscheidungsprozesses bietet sich daher die Fuzzy–Logik in Form eines Expertensystems an. Vorteile sind eine bessere Transparenz und damit Akzeptanz durch den einzelnen Planer, im Vergleich zu Optimierungslösungen mit den notwendigen, die Optimallösung beeinflussenden Kostenannahmen. Da die Fuzzy–Logik zu den qualitativen Planungsmethoden gehört, besitzt diese einen heuristischen Charakter, so dass nicht zwangsläufig eine Konvergenz zu einer optimalen Lösung gegeben ist. Nichtsdestotrotz bietet sich die Entwicklung eines Entscheidungssystems durch Formalisierung von heuristischen Expertenregeln an, um den entsprechenden Logistikplaner durch Vorschlagsunterbreitung und eine standardisierte Informationsversorgung zu unterstützen.

Ziel der Fuzzy–Logik [6] ist es, toleranzbehaftete Aussagen automatisiert mittels eines mathematischen Systems verarbeiten zu können. Die Grundproblematik scharfer Grenzen in vielen Anwendungsfällen kann durch die Fuzzy–Logik vermieden werden. Wissen kann in Form von Fuzzyregeln vom Typ Mamdani [7] strukturiert und verarbeitet werden, indem Fuzzymengen in den Wenn–Dann–Regelstrukturen verwendet werden.

4.2 Festlegung der Kriterien und Regeln

In einem Brainstorming wurden mögliche Segmentationskriterien entwickelt. Das Resultat zeigte, dass folgende Kriterienklassen berücksichtigt werden sollten:

1. Produkt-/produktionsrelevante Faktoren (z.B. Produktlebenszyklusphase, Variantenbildungsgrad im Materialfluss),
2. marktrelevante Faktoren, wie Stärke der Nachfrageschwankungen,
3. kosten- und gewinnorientierte Faktoren (wie ABC–Klassifikation des Umsatzbeitrags),
4. weitere, spezielle kundenrelevante Faktoren.

Um im Interview mittels einer strukturierten Vorgehensweise eine möglichst konsistente Regelbasis zu entwickeln, wurden fünf Hauptkriterien ausgewählt, deren vollständige Kombinationen ausformuliert wurden. Die resultierenden 243 Möglichkeiten der Kombination der linguistischen Variablen der jeweiligen Eingangsvariablen wurden dann mit einem Service Level Code als Entscheidungsempfehlung versehen.

Nach interner Abstimmung wurden problematische Entscheidungen identifiziert. Diese Problemfälle wurden zusammengestellt und in einem Expertenteam durchgesprochen. Auf dieser Basis dieser Expertenrunde wurde jeweils nach Diskussion ein Service Level Code vergeben.

4.3 Implementierung und Anwendung

Die informationstechnische Umsetzung der dargestellten Entscheidungsfindung erfolgte mit Hilfe zweier Microsoft Access Datenbanken und wurde in SQL in Verbindung mit Visual Basic for Applications programmiert. Nach Betriebsdatenerfassung aus den Bereichen Produktgrunddaten, Kostendaten und Daten der Auftragshistorie werden diese in die erste Datenbank eingelesen und dort verarbeitet. In dieser Analyse–Datenbank werden die notwendigen Berechnungsschritte durchgeführt, um produktbezogen die einzelnen Kriterienausprägungen zu ermitteln. Diese Werte sind grö"stenteils bisher nicht in dieser Form automatisiert abrufbar und müssen daher erst ermittelt und in eine Datenquelle zusammengeführt werden.

Eine zweite Datenbank übernimmt diese Werte und führt die beschriebenen Schritte aus. Diese Datenbank überführt die scharfen Kriterienausprägungen in Fuzzy–Werte, berechnet die Regelaktivierungsgrade und wählt den höchsten Regelaktivierungsgrad aus. Mit Hilfe der neuen Segmentationsliste soll der Forderung nach Transparenz der Entscheidungsfindung nachgekommen werden, indem die einzelnen (scharfen) Kriterien zusätzlich pro Endprodukt angegeben werden. Zusätzlich wird die ausschlaggebende Regel jeweils mit angezeigt. Die Analysedatenbank erlaubt weiterhin Auswertungen bezüglich der Veränderung der Segmentierungsmatrix, die Auswertung der prozentualen Änderungen je Geschäftsgebiet und eine Anzeige einer Regelstatistik.

5 Zusammenfassung und Schlussfolgerungen

Der Inhalt des Order Penetration Point Konzeptes und der logistischen Produktionsprinzipien ist nicht neu und findet sich schon in der Literatur zur Entwicklung von Betriebstypologien, beispielsweise bei [8]. Die Festlegung des Order Penetration Point stellt dabei ein äußerst wichtiges Element dar bei dem Übergang von einer reinen Produktionsplanung und -steuerung zu einem aktiven Auftragsmanagement. Dabei ist besonders der Einhaltung des Kundenwunschtermins und der dafür vom Kunden zugestandenen Lieferzeit verstärkte Beachtung zu widmen. Das Konzept der neuen Produktionssystematik bezieht diese Kriterien ein, die Lagerbestandsplanung mit Fokus auf eine stärkere Kun-

denorientierung wird so verbessert. Umgesetzt werden konnte die Festlegung der Bevorratungsstufe mit Hilfe von Kriterien, die über eine reine ABC-orientierte Entscheidung hinausgehen. Ferner ist festzuhalten

- Das Framework von Olhager konnte erfolgreich in ein praktisch anwendbares Entscheidungsunterstützungssystem umgesetzt werden.
- Fuzzy–Logik hat sich als gutes mathematisches Instrumentarium bei der Entscheidungsfindung erwiesen.
- Die Berücksichtigung der vom Kunden gewünschten Lieferzeit als Entscheidungskriterium zur Festlegung der Bevorratungsebene konnte umgesetzt werden.
- Ein strukturierter systematischer Entscheidungsvorschlag kann innerhalb kurzer Zeit erstellt werden.

Die Reaktionen der Geschäftsbereiche sind sehr positiv ausgefallen. Die in ihrer Kompaktheit zur Verfügung gestellte Produktinformationen auf der niedrigen Aggregationsstufe wurde sehr begrü"st. Die systematischen Vorschläge zur Festlegung der Bevorratungsebene sind insgesamt von den Logistikplanern positiv aufgenommen und in den überwiegenden Fällen auch umgesetzt worden. Die strukturierte Bestimmung der Bevorratungsebene ist bei ca. 75% der betrachteten Produkte möglich.

References

1. Rajagopalan, S.: "Make to Order or Make to Stock: Model and Application" In: Management Science, 2002, 2:241-256.
2. Schömig, A., Fowler, J.W.: "Modelling Semiconductor Manufacturing Operations." In: *Proceedings of the 9th ASIM Dedicated Conference Simulation in Production und Logistics.* Berlin, March 2000, 56–64.
3. Stadtler, H., Kilger, C. (Eds.) Supply Chain Management and Advanced Planning. Concepts, Models, Software and Case Studies. Berlin 2000.
4. Olhager, J.: " Strategic positioning of the order penetration point" In: *International Journal of Production Economics*, 2003, 3: 319–329.
5. Sharman, G.: "The rediscovery of logistics" In: *Harvard Business Review*, 1984, 5:71–79.
6. Zadeh, L. A.: "Fuzzy Sets" In: *Information and Control*, 1965, 3:338–353.
7. Durkin, J.: Expert Systems: Design and Development, New York 1994: 60.
8. Giesberts, P. M. J.; v. d. Tang, L.: "Dynamics of the customer order decoupling point: impact on information systems for production control" In: *Production Planning* & Control, Band 3, 1992, 3:300-313.

On the Value of Objective Function Adaptation in Online Optimisation

Jörn Schönberger and Herbert Kopfer

University of Bremen, Chair of Logistics, Wilhelm-Herbst-Straße 5,
28359 Bremen, Germany. {jsb,kopfer}@uni-bremen.de

Summary. We analyse a dynamic variant of the vehicle routing problem with soft time windows in which an average punctuality must be guaranteed (e.g. lateness is allowed at some customer sites). The existing objective function does not support both the aspiration for punctuality and least cost so that additional efforts are necessary to achieve an acceptable punctuality level at least possible costs. Within numerical experiments it is shown that static penalties are not adequate in such a situation but that an adaptation of the objective function before its application to the next problem instance supports the search for high quality solutions of the problem.

1 Introduction

We consider a vehicle routing problem in which the adequate fulfilment mode of consecutively arriving customer requests is to be selected. Arriving requests are fulfilled using the cheap but not necessarily reliable self-fulfilment mode (SF) or the more expensive but reliable subcontracting mode (SC). We propose to adaptively adjust the weights of the costs of the two fulfilment modes in a mono-criterion objective function. Doing so, we refine the idea of Gutenschwager et al. [1] who initially propose to adjust higher-ranked objective functions to the recent decision situations while solving the next instance in a sequence of optimisation models in online-fashion. This adaptive approach is compared in numerical experiments with the typically used penalty approach in which a static unchangeable value is used to depreciate late visits at customer sites.

Section 2 introduces the investigated decision problem. Section 3 outlines the decision algorithms. The computational experiments are reported in Section 4.

2 Dynamic Decision Problem

We investigate the following generalisation of the vehicle routing problem with time windows.

Soft time windows. Lateness at a customer site is possible but causes penalty costs. The portion p_t of the requests completed or scheduled for completion in the interval $[t-t^-;t+t^+]$ is observed. At least p_t of all requests of the interval around the current time t must be started within the agreed time windows.

Subcontraction. Each request can be served by a vehicle from the own fleet in the SF-mode or it can be subcontracted (SC-mode) to a logistic service provider (LSP). In the former case, late arrivals at customer sites cannot be prevented but in the latter case, an in-time service is assured. A once subcontracted request cannot be re-integrated into the routes of the own vehicles.

Uncertain demand. Only a subset of all requests is known to the planning authority at the time when the decision concerning subcontracting is made and the routes for the own vehicles are generated. The planning authority decides about the fulfilment mode of a request as soon as it becomes known.

Additional requests arriving at time t_i trigger the update of the so far followed transportation plan TP_{i-1} which contains the decision how the waiting requests will be served. For the update of TP_{i-1} to TP_i we have introduced an optimisation model [2] whose solving identifies a least costs refresh of the transportation plan. Each possible update is evaluated by the resulting costs using the objective function (1).

$$\underbrace{C_1(\mathcal{RP}(\mathcal{R}_i^{int})) + C_2(\mathcal{RP}(\mathcal{R}_i^{int}))}_{\text{self-fulfilment costs}} + \underbrace{C_3(\mathcal{SC}(\mathcal{R}_i^{ext}))}_{\text{SC usage costs}} \rightarrow \min . \qquad (1)$$

The set $\mathcal{R}_i^{int}$ contains all requests for which the self-fulfilment mode has been selected and the set $\mathcal{R}_i^{ext}$ comprises all subcontracted requests at time t_i. With $\mathcal{RP}$, we denote the least cost collection of paths for the own vehicles and $\mathcal{SC}$ refers to the minimal-charge bundling of subcontracted requests. Then, $C_1(\mathcal{RP}(\mathcal{R}_i^{int}))$ denotes the travel costs of the own vehicles, $C_2(\mathcal{RP}(\mathcal{R}_i^{int}))$ gives the penalty costs to be paid for late customer site visits of own vehicles. Finally, $C_3(\mathcal{SC}(\mathcal{R}_i^{ext}))$ gives the costs of the subcontracted requests.

If both fulfilment modes would lead to the same costs for a given request r, e.g. if

$$\alpha := \frac{C_3(TP3(r))}{C_1(TP1(\mathcal{R}_i^{int})) + C_2(TP1(r))} \approx 1, \tag{2}$$

then both fulfilment modes SF and SC will be used to the same extend as long as the limited capacity of the own fleet is exhausted. As soon as the capacity of the own fleet is exhausted then some requests are shifted into the SC-mode. However, if $\alpha \gg 1$ then the aspiration for cost minimal modes prevent the usage of the SC mode. If C_2 is not stringent and severe enough, then the number of late severed requests increases, so that p_t falls down.

In the remainder of this article we investigate the dependencies between the severeness of the penalisation of late requests and the selection of the fulfilment mode. Thereby, we assume that $\alpha \gg 1$, so that the aspiration for least cost transportation plans does not support the selection of the mode that leads to the highest percentage of punctually served requests.

We use artificial test cases [2] constructed from the 100-customer Solomon [3] instances $\{R103, R104, R107, R108\}$ for an experimental analysis of the aforementioned situation. In these scenarios, a demand peak leads to a temporal exhaustion of the cheaper SF mode. We propose and test ideas to overrule the cost-based mode decision in order to consider punctuality issues to a larger extend.

3 Algorithm Details

We use the Memetic Algorithm described e.g. in [2] to derive a new transportation plan after additional requests have arrived. In such a case, the execution of TP_{i-1} is interrupted and TP_{i-1} is replaced by TP_i.

A piece-wise linear penalty function h is deployed, which is 0 for delays up to T_{max} time units and which increases proportionally up to a maximal penalty value P_{max} (money units) for delays longer than the threshold delay of 100 time units. Using this penalty calculation the sum of penalty payments is $C_2(\mathcal{RP}(\mathcal{R}_i^{int})) := \sum_{r \in \mathcal{R}^{int}} h(delay(r))$, where $delay(r)$ gives the distance to the latest allowed visiting time at the customer site corresponding to request r.

The previously introduced penalty function h is deployed with different parameter settings. We perform simulations with the maximal penalty values $P_{max} \in \{50, 75, 100, 125\}$ and the tolerance ranges $T_{max} \in \{0, 25, 50, 75\}$. A parameterisation of the penalty function is denoted by $P(P_{max}, T_{max})$.

Alternatively, we deploy an adaptation mechanism that re-weights the costs of the two fulfilment modes in the objective function in dependence from the currently observed punctuality p_t. The idea of this approach is to artificially lower the costs of the SC mode (compared to the SF mode) if p_t is low in order to make the usage of the SC mode more attractive.

$$f(t_i) \cdot \left[C_1(\mathcal{RP}(\mathcal{R}_i^{int})) + C_2(\mathcal{RP}(\mathcal{R}_i^{int}))\right] + C_3(\mathcal{SC}(\mathcal{R}_i^{ext})) \rightarrow \min \quad (3)$$

The coefficient $f(t_i)$ is adjusted before the update of TP_{i-1} to TP_i starts. It is $f(t_0) = 1$ and $f(t_i) = 1 + \alpha \cdot \Omega(t_i, p_{t_i})$ for $i \geq 1$. We use the piece-wise linear function Ω which is 0 if the current punctuality p_{t_i} is larger than $p^{target} + 0.05$ and which equals 1 if $p_{t_i} \leq p^{target} - 0.05$. In the latter case, it is $f(t_i) = 1 + \alpha$ and subcontracting a request is identified by the solver via the objective function to be cheaper than the self-fulfilment with respect to the currently used objective function (3). For p_t-values between $p^{target} - 0.05$ and $p^{target} + 0.05$ the function Ω decreases proportionally from 1 down to 0. Since the re-definition of the coefficient affects the search trajectory heading of the solving algorithm, we call this approach Search Direction Adaptation (SDAD).

4 Numerical Experiments

Experimental Setup. We analyse two scenarios. In scenario I, SF and SC have the same prices ($\alpha = 1$) but in scenario II SC is quite more expensive than SF ($\alpha = 3$).

A single simulation run $(P, \omega, \epsilon, \alpha)$ is determined by the request set $P \in \{R103, R104, R107, R108\}$, the algorithm seeding $\omega \in \{1, 2, 3\}$, the applied strategy $\epsilon \in \{SDAD\} \cup \{PEN(a, b) \mid a \in \{50, 75, 100, 125\}, b \in \{0, 25, 50, 75\}\}$ and $\alpha \in \{1, 3\}$. Thus, $4 \cdot 3 \cdot 17 \cdot 2 = 406$ simulation runs have been executed.

Throughout the simulations we observed the maximal punctuality decrease (in percent) $\delta(\epsilon, \alpha)$ after the demand peak and the cumulated overall costs $C(\epsilon, \alpha)$.

The results observed for the PEN$(\cdot, \cdot)$-experiments in scenario I are presented in Fig. 1. The left isoline-plot shows the observed maximal punctuality decreases $\delta(\alpha, \epsilon)$. In $A_1^{\delta}(-0.6)$ maximal punctuality variations between -0.6% and 0 (light grey shaded area) are observed. Punctuality variations between -1% and -0.6% appear in $A_1^{\delta}(-1)$. Decreases of p_t between 1% and 1.4% take place in $A_1^{\delta}(-1.4)$

The right isoline plot compiles the average of the cumulated costs $C(\cdot,\cdot)$ occurred during the simulation runs within the PEN$(\cdot,\cdot)$-experiments. Additional costs of less than 5% ($A_1^C(0)$) are observed for small penalties and high tolerance values (light grey shaded) areas. A cost increase of more than 15% is realized if $T_{max} \leq 25$ and $P_{max} \geq 75$.

The application of SDAD leads to a maximal punctuality decrease of 1% at nearly the same costs (dark grey shaded areas in the two plots). We conclude, that if $\alpha = 1$ (same costs for SF and SC) then the static parameter setting $(50, 75)$ of h performs sufficient with respect to a sufficiently high service quality as well as service cost minimisation.

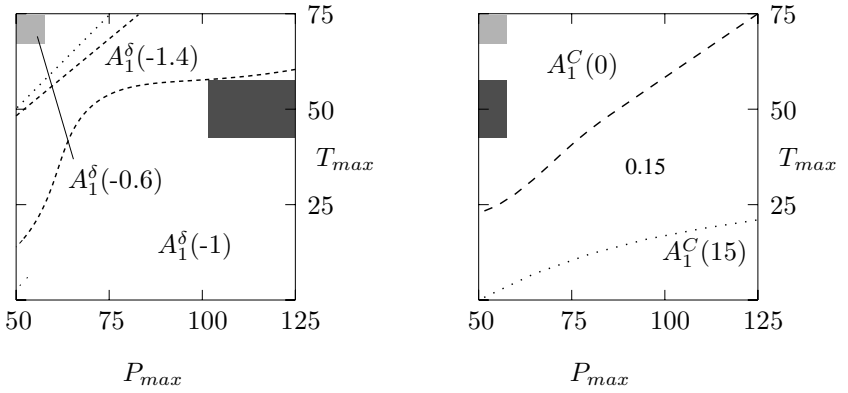


Fig. 1. Scenario I ($\alpha = 1$): punctuality decrease (left) and costs (right)

Quite different results are observed in scenario II (Fig. 2). The service quality optimal parameter setting is $P_{max} = 125$ and $T_{max} = 25$ with a maximal punctuality reduction of 2.9% at costs of 72284,10 (light grey shaded area in the left plot in Fig. 2). This setting causes additional costs of more than 15% (cf. right plot in Fig. 2). On the other hand, the cost optimal parameter setting ($P_{max} = 50, T_{max} = 75$, (light grey shaded area in the right plot in Fig. 2) results in a punctuality collapse of around 20%. It is therefore not possible to find a parameter setting for h that satisfies both goals costs minimisation and punctuality preservation to the maximal extend at the same time.

For both reasonable tradeoff parameter settings (100,50) and (75,50) we observe a significantly higher punctuality decrease (compared to the punctuality preserving setting) or quite enlarged costs (compared to the cost optimal setting).

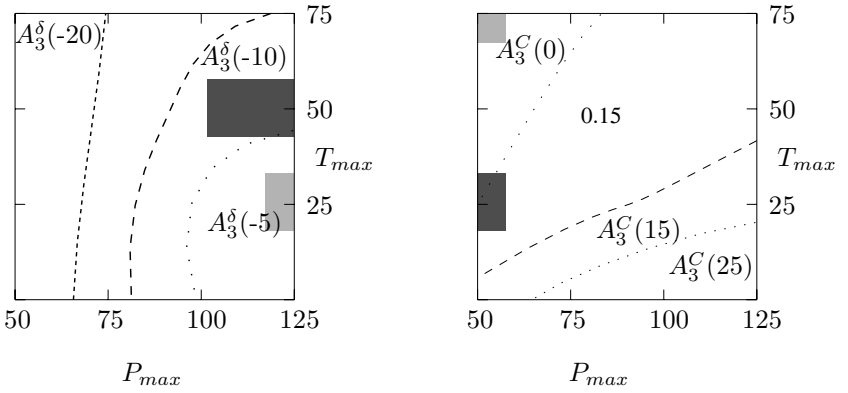


Fig. 2. Scenario II ($\alpha = 3$): Punctuality decrease (left) and costs (right)

In contrast, SDAD performs very well. It comes along with an acceptable punctuality decrease of only 5.7% which is better than the performance of the two trade-off proposals (dark gray shaded areas in Fig. 2). The costs resulting from the application of SDAD are only 63888,7 which is a significant reduction of the costs compared to the two proposed trade-off parameter settings.

5 Conclusions and Outlook

We have shown that the adaptive definition of an objective function supports the achievement of a good trade-off between service quality and service costs in a volatile environment. Further research activities are dedicated the identification of the right key indicators to be used to derive the right adaptation decisions.

References

1. K. Gutenschwager, F. Böse, and S. Voß. Effiziente Prozesse im kombinierten Verkehr – ein neuer Lösungsansatz zur Disposition von Portalkränen. *Logistik Management*, 5(1):62–73, 2003.
2. J. Schönberger and H. Kopfer. Flexible transport process planning in volatile environments and the adaptation of a cost-based objective function. In *Proceedings of Logistik Management 2007*, pages 263–283. Gabler-Verlag, 2007.
3. M. Solomon. Algorithms for the vehicle routing and scheduling problem with time window constraints. *Operations Research*, 35(2):254–265, 1987.

A Novel Multi Criteria Decision Making Framework for Production Strategy Adoption Considering Interrelations

Nima Zaerpour, Masoud Rabbani, and Amir Hossein Gharehgozli

Department of Industrial Engineering, University of Tehran, P.O. Box: 11155/4563, Tehran, Iran. nzaerpour@ut.ac.ir

Summary. Firms produce some of their products utilizing Make-To-Order (MTO), some utilizing Make-To-Stock (MTS) another some Hybrid system.The present study examines a novel hybrid method for improving the usability of SWOT (Strengths, Weaknesses, Opportunities and Threats) and ANP (Analytic Network Process). SWOT analysis is always carried out regarding environmental and internal criteria. Although outer dependence of corresponding factors is inevitably taken into account, inter-dependence among the factors is seldom addressed. This criterion makes ANP more powerful than AHP . However, due to the complexity and uncertainty involved in real world decision problems, it is sometimes unrealistic or even impossible to require exact judgments. Therefore Fuzzy Analytic Network Process (FANP) are integrated with SWOT analysis.

Key words: Make-To-Order; Make-To-Stock; SWOT analysis; Analytic Network Process; Fuzzy sets theory.

1 Introduction

A manufacturing system can be defined as an arrangement of tasks or processes to transform a selected group of raw materials or semi-finished products into a set of finished products. From the viewpoint of the relationship between production release and order arrival, production systems can be classified into Make-To-Stock (MTS) and Make-To-Order (MTO) or Hybrid one. For a make-to-stock system, finished or semi-finished products are produced to stock according to forecasts of the demands. In a make-to-order system, work releases are authorized only in accordance with the external demand arrival.

The literature on the issue of MTS versus MTO goes back to the 1960s when Popp [1] proposed a simple single-item stochastic inventory model with zero lead-time for production or replenishment of an item. Simple cost comparisons of making the item to stock versus making it to order were presented. Sox et al. [2] also addressed a similar problem where some items are MTS and some MTO, with demand for MTO items to be satisfied within a certain time interval. They considered more complex scheduling policies.

In recent years, Rajagopalan [3] has suggested a non-linear integer program with service level constraints and heuristic procedures to solve MTS/MTO partitioning problem. Soman et al. [4] have discussed about a comprehensive hierarchical planning framework that covers the important production management decisions to serve as a starting point for evaluation and further research on the planning system for MTO/MTS situations.

In this study, a new methodology, "Fuzzy ANP-SWOT" is proposed to decide whether an item should be produced in MTO, MTS or Hybrid MTS-MTO environment. The combination of Analytic Network Process with SWOT analysis is a novel approach in strategic decision-making process.

2 Identifying Product, Firm and Process Considering External and Internal Environments

At first characteristics of product, firm and process considering internal and external factors which directly affect on the strategy chosen for producing the product should be identified. These factors are as follows: *Product-related criteria* (Cost of each item, Risk of obsolescence and perishability, etc.) and *firm and process related criteria* (Human resource flexibility, Equipment flexibility, Delivery lead-time, Return of investment, Customer commitment, Supplier commitment, etc.).

3 Application of Fuzzy ANP-SWOT Methodology

The first step involves identifying key factors that influence the decision (SWOT analysis is carried out). So Prior to the application of ANP which is a Multi Criteria Decision Making (MCDM) technique, Factors in each group of SWOT should be identified. The steps of proposed methodology are as follows:

Step 1. SWOT analysis is carried out: The relevant factors of the external and internal environment are identified and included in SWOT analysis with considering mentioned criteria.

Step 2. Identification of network structure and its relationship: In this step, we should construct a network with Strength, Weakness, Opportunity, and Threat at the top level. They are classified in a category, which is named determinants. In the ANP model, the higher the level, the more encompassing or strategic the criteria. Hence, SWOT form the upper level of the ANP model. Each determinant includes some factors or criteria derived from SWOT analysis. In the network, all outerdependence and interdependence should be illustrated

Step 3. Pair-wise comparison of determinants: In this step, a series of fuzzy pair-wise comparisons are made to establish the relative importance of determinants in achieving the objective.

Step 3.1. Establish fuzzy judgment matrix: This matrix represents the relative performance (importance) of determinants. It is not possible to make mathematical operations directly on linguistic values. This is why; the linguistic scale must be converted into a fuzzy scale. Fuzzy judgment matrix can be expressed as: $\tilde{A} = (\tilde{A}_{ij})_{n\times n}$.

Step3.2. Establish the total fuzzy judgment matrix with $\alpha - cut$ and the β degree of satisfaction of experts on judgment: This matrix represents the degree of satisfaction of the experts on the judgment. In order to represent the degree of the optimism of a decision maker, α should be fixed and then the index of optimism β can be set. A larger β indicates a higher degree of optimism, and vice versa. The index of optimism is a linear convex combination and is defined as: $A^{\beta}_{ij\alpha} = (1-\beta)A^{l}_{ij\alpha} + \beta A^{u}_{ij\alpha}$. Thus the total fuzzy judgment matrix with $\alpha - cut$ and index of optimism β leads to the crisp pair-wise comparison matrix which we use in the next steps. It can be expressed as: $A = (A^{\beta}_{ij\alpha})_{n\times n}$.

Step 4. Alternative methods for weights calculation

Step 4.1. Eigenvalue method: Matrices are not always perfectly consistent and contain inconsistency. When contains inconsistencies, the estimated priorities can be obtained by using the eigenvalue technique:

$$(A^{\beta}_{\alpha} - \lambda_{max} I)W = 0 \tag{1}$$

Where λ_{max} is the largest eigenvalue of matrix A^{β}_{α}, I is the identity matrix, and W constitutes the estimation relative priorities.

Saaty [5] has shown that λ_{max} of a reciprocal matrix A is always greater or equal to n (number of rows or number of columns). The more consistent the comparisons are, the closer the value of computed

λ_{max} is to n. Based on this property, a consistency index, CI, has been constructed which is reflected in the following equation: $CI = \frac{\lambda_{max}-n}{n-1}$.

CI estimates the level of consistency with respect to a comparison matrix. Then, because CI is dependent on n, a consistency ratio CR is calculated which is independent of n ($CI = \frac{CI}{RCI}$). It measures the coherence of the pair-wise comparisons. To estimate CR, the average consistency index of randomly generated comparisons, RCI, has to be calculated. RCI varies functionally, according to the size of the matrix. As a rule of thumb, a CR value of 10% or less is considered acceptable.

Step 4.2. An approximate method: Since using eigenvalue method is rather difficult, In order to find the priorities from the pair-wise comparison matrix, we use an alternative method:

$$\psi_j = \sum_{i=1}^{n} A^{\beta}_{ij\alpha} \qquad \forall j \tag{2}$$

$$A^{\beta}_{\alpha norm} = (\frac{A^{\beta}_{ij\alpha}}{\psi_j})_{n\times n} \tag{3}$$

$$\frac{\sum_{j=a}^{n} A^{\beta}_{ij\alpha}/\psi_j}{n} = w_i \qquad \forall i \tag{4}$$

Step 5. Fuzzy Pair-wise comparisons of SWOT factors: These factors in each SWOT group are compared and their relative priorities are calculated as in Steps 3, 4. In this step, the fuzzy pair-wise comparison of factors at each level is conducted with respect to their relative influence towards their control criterion.

Step 6. Fuzzy Pair-wise comparison matrices for interdependencies: In this step, pairwise comparisons are made to capture interdependencies among the factors. For example suppose in Opportunity group, O_1 is controlling factor over other factors. In the formation of this table, the question asked to the decision-maker is: "when considering O_1 with regard to Opportunity determinant, what is the relative importance of factor O_2 when compared to factor O_3?"

Step7. Fuzzy Pair-wise comparison of Alternatives: The final set of pair-wise comparisons is made for the relative impact of each of the alternatives (MTS, MTO, and Hybrid) on the factors in influencing the determinants.

Step8. Super-matrix formation: The super-matrix allows for a resolution of interdependencies that exist among the factors. The super-matrix M, presents the results of the relative importance measures for

each of the factors. The elements of the super-matrix have been imported from the pair-wise comparison matrices of interdependencies. In the next stage, the super-matrix is made to converge to obtain a long-term stable set of weights. For convergence to occur, the super-matrix needs to be column stochastic. Convergence happens when there is no significance difference between matrix elements of two consecutive steps.

Step 9. Calculation of Overall Weighted Index (OWI) and selection of the best alternative: Selection of the best alternative depends on the values of Overall Weighted Index for each alternative. These values are computed by [6]: $OWI_i = \sum_{a=1}^{A} \sum_{k=1}^{k_a} C_a A_{ka}^D A_{Ka}^I S_{ika}$.

In this equation C_a is the relative importance weight of determinant a which is computed in step 3. A_{ka}^D is the relative importance of factor k in influencing determinant a for the dependency (D) relationships and is derived from step 5. A_{ka}^I is stabilized (after convergence) importance weight of the factor k in the determinant a cluster for interdependency (I) relationships. This parameter is calculated through steps 6 and 8. S_{ika} is the relative impact of alternative i on factor k for determinant a. The number of determinants is denoted by A and the number of relative factors of each determinant is symbolized by K_a. The summation of parameters' product for each alternative provides the value of Overall Weighted Index of corresponding alternative (OWI_i). OWI_i shows the total weight of each alternative (MTS, Hybrid MTS/MTO and MTO) and as result the alternative with greater OWI is chosen as a production strategy for the new product.

4 Illustrative Case Study

This case concerns a food processing company that wants to make a new product. The question is which production strategy should be adopted to produce the new product? Make-to-Order, Hybrid MTS/MTO or Make-to-Stock? The project team members analyzed all data; they finally came into three SWOT factors in each group which affect the decision on MTS or MTO or Hybrid MTS/MTO application.

By applying Steps 1-9 in Section 2, the performance of different producing systems under different level of α and optimism index β are calculated and shown in Figure 1. The results are variant depending on the values of α and β. In the following figure the number 0 on the vertical axis indicates that we should employ MTS system, the number 1 indicates that we should employ Hybrid MTS-MTO and the number 2 indicates that we should employ MTO system.

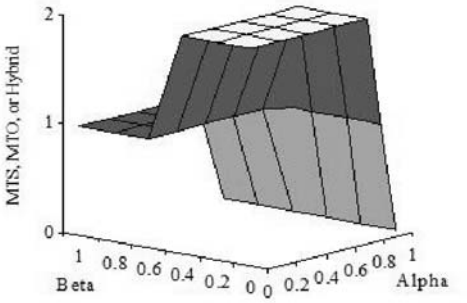


Fig. 1. The performance of different producing systems under different level of α and optimism index β

5 Conclusion

In this study, a novel hybrid methodology is utilized in MTO/MTS manufacturing systems for partitioning the MTO/MTS products. Making decision on producing an item under MTS, Hybrid MTS/MTO or MTS strategy in such companies is an important and strategic process. Fuzzy ANP-SWOT approach is proposed as a strategic decision-making methodology for partitioning the products.

References

1. Popp W (1967) Simple and combined inventory policies, production to stock or to order? Management Science, 11(9): 868-873
2. Sox CR, Thomas LJ, McClain JO (1997) Coordinating production and inventory to improve service. Management Science, 43(9): 1189-1197
3. Rajagopalan S (2002) Make-to-order or make-to-stock: Model and application. Management Science, 48(2): 241-256
4. Soman CA, Van Donk DP, Gaalman G (2004) Combined make-to-order and make-to-stock in a food production system; International Journal of Production Economics 90: 223-235
5. Saaty TL (1977) A scaling method for priorities in hierarchical structure. Journal of Mathematical Psychology, 15: 234-281
6. Meade LM Sarkis J (1999) Analyzing organizational project alternatives for agile manufacturing processes: An analytic network approach. International Journal of Production Research, 37(2): 241-261