Lecture Notes in Economics and Mathematical Systems 682

Raquel J. Fonseca
Gerhard-Wilhelm Weber
João Telhada *Editors*

Computational Management Science

State of the Art 2014

Springer

Lecture Notes in Economics and Mathematical Systems 682

More information about this series at
http://www.springer.com/series/300

Raquel J. Fonseca • Gerhard-Wilhelm Weber •
João Telhada
Editors

Computational Management Science

State of the Art 2014

Springer

Editors
Raquel J. Fonseca
Faculty of Sciences/Operations Research Center
University of Lisbon
Lisbon, Portugal

João Telhada
Faculty of Sciences/Operations Research Center
University of Lisbon
Lisbon, Portugal

Gerhard-Wilhelm Weber
Institute of Applied Mathematics
Middle East Technical University
Ankara, Turkey

ISSN 0075-8442 ISSN 2196-9957 (electronic)
Lecture Notes in Economics and Mathematical Systems
ISBN 978-3-319-20429-1 ISBN 978-3-319-20430-7 (eBook)
DOI 10.1007/978-3-319-20430-7

Library of Congress Control Number: 2015957188

Springer Cham Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer International Publishing AG Switzerland is part of Springer Science+Business Media (www.springer.com)

Preface

Computational Management Science has become an important research field, in particular in recent years due to the rapid growth of computational power and data access, storage, and management. These facts are shaping modern management, highlighting analytical issues that raise added value to business, and in general to the economy.

With the background of increasing importance of the management science field, namely with concern to the computational aspects, a series of annual conferences started to be held in 2004.

The 2014 edition took place in the beautiful and inspiring city of Lisbon, Portugal. The conference chairs (Raquel J. Fonseca, Daniel Kuhn, and João Telhada) chose Energy and Finance as the two major topics to be discussed, due to the global revelancy of any of those two areas. The response by the research community was worth noting, as 125 valid submissions were received. Out of those 125 submissions, 99 were accepted for presentation in the conference. Those submissions were divided among nine streams, according to the most adequate topic.

Participants in the conference came from a total of 22 countries, spanning different regions as far Asia or South America. Countries with the largest number of participants were Portugal (31) and the United Kingdom (14).

The organizing committee was composed by the following members:

- **Ana Luísa Respício**, Operations Research Center, University of Lisbon
- **Inês Marques**, Operations Research Center, University of Lisbon
- **João Patrício**, Telecommunications Institute, Polytechnique Institute of Tomar
- **João Telhada** (co-chair), Operations Research Center, University of Lisbon
- **Lisete Sousa**, Center of Statistics and Applications, University of Lisbon
- **Luís Gouveia**, Operations Research Center, University of Lisbon
- **Miguel Constantino**, Operations Research Center, University of Lisbon
- **Raquel João Fonseca** (co-chair), Operations Research Center, University of Lisbon

A group of specialized researchers composed the Program Committee to ensure a high scientific standard among submissions. That committee had the following members:

- **A. Ismael F. Vaz**, Universidade do Minho, Portugal
- **Aharon Ben-Tal**, Technion – Israel Institute of Technology, Israel
- **Alain Haurie**, Université de Genève, Switzerland
- **Alex Weissensteiner**, DTU – Technical University of Denmark, Denmark
- **Ana Luísa Custódio**, Universidade Nova de Lisboa, Portugal
- **Ana Paula Barbosa Póvoa**, Universidade de Lisboa, Portugal
- **Berç Rustem**, Imperial College London, United Kingdom
- **Bernardo Almada Lobo**, Universidade do Porto, Portugal
- **Carlos Henggeler Antunes**, Universidade de Coimbra, Portugal
- **Celso Ribeiro**, Universidade Federal Fluminense, Brazil
- **Christodoulos Floudas**, Princeton University, United States
- **Cristian Gatu**, Université de Neuchâtel, Switzerland
- **Dan Iancu**, Stanford University, United States
- **Daniel Kuhn** (chair), École Polytechnique Fédérale de Lausanne, Switzerland
- **David Wozabal**, Technische Universität München, Germany
- **Diogo Pinheiro**, Brooklyn College of the City University of New York, United States
- **Erricos Kontoghiorghes**, Queen Mary University of London, United Kingdom
- **Filipe Alvelos**, Universidade do Minho, Portugal
- **Georg Pflug**, University of Vienna, Austria
- **Georges Zaccour**, HEC Montréal, Canada
- **Gerhard-Wilhelm Weber**, Middle East Technical University, Turkey
- **Giorgio Consigli**, Università di Bergamo, Italy
- **Huifu Xu**, City University London, United Kingdom
- **Jacques Savoy**, Université de Neuchâtel, Switzerland
- **João Carlos Lourenço**, Universidade de Lisboa, Portugal
- **Joaquim Júdice**, Universidade de Coimbra, Portugal
- **José Fernando Oliveira**, INESC TEC, Portugal
- **Laureano Escudero**, Universidad Rey Juan Carlos, Spain
- **Luis Seco**, University of Toronto, Canada
- **Marco Campi**, University of Brescia, Italy
- **Marcus Poggi**, Pontifícia Universidade Católica do Rio de Janeiro, Brazil
- **Maria Antónia Carravilla**, Universidade do Porto, Portugal
- **Miguel Gomes**, Universidade do Porto, Portugal
- **Miguel Lejeune**, The George Washington University, United States
- **Mustafa Pinar**, Bilkent University, Turkey
- **Nalan Gulpinar**, University of Warwick, United Kingdom
- **Panos Pardalos**, University of Florida, United States
- **Panos Parpas**, Imperial College London, United Kingdom
- **Patrícia Xufre**, Universidade Nova de Lisboa, Portugal

- **Paula Bouzas**, Universidad de Granada, Spain
- **Pedro Martins**, Polytechnic Institute of Coimbra, Coimbra Business School, Portugal
- **Peter Winker**, Universität Giessen, Germany
- **Ronald Hochreiter**, WU Vienna University of Economics and Business, Austria
- **Spiros Martzoukos**, University of Cyprus, Cyprus
- **Stein-Erik Fleten**, Norwegian University of Science and Technology, Norway
- **Teemu Pennanen**, King's College London, United Kingdom
- **Tomasz Radzik**, King's College London, United Kingdom
- **Werner Römisch**, Humboldt-Universität zu Berlin, Germany
- **Wolfram Wiesemann**, Imperial College London, United Kingdom

The conference scientific programme included four distinguished keynotes by notorious researchers in the field of Management Science. The speakers and titles were the following:

- **Victor DeMiguel**, London Business School—*Data Driven Investment Management*
- **William Pulleyblank**, United States Military Academy, West Point—*Analytics, Sports and Force-on-Force Situations*
- **Daniel Ralph**, University of Cambridge, Judge Business School—*Capacity decisions in electricity production under risk aversion and risk trading*
- **Rüdiger Schultz**, Zentrum für Logistik & Verkehr, Universität Duisburg-Essen—*From One to Infinity - Dimensions of Stochastic Programming*

This set of outstanding talks gave an excellent motivation for the remainder of the conference, not only due to the excellency of communication, but also because it gave a thorough outlook into the main topics of the conference, thus setting the tone for all scientific interaction.

Following the success of the conference, a set of short papers were submitted for publication. A total of 49 contributions were presented for appreciation by the reviewers. After each paper was evaluated by no less than 2 reviewers, 32 of those papers were selected to contribute and to compose this volume.

Contributions were arranged in terms of major application areas of computational management science. Some, however, focus more on optimization methods and, therefore, fit in a separate section.

The first part includes contributions on Energy. Quintino et al. apply a holistic approach to the problem of hedging in energy market against oil and gas prices volatility by confronting the performance of a portfolio for the entire company with that of the sum of each business unit. A portfolio optimization model is also applied to the decision of choosing both demand and its flexibility by Gärttner et al., but in the context of a renewable energy firm that needs to meet its demand with a volatile supply. Mudry and Paraschiv focus on the stress testing of a portfolio of commodity futures, and the impact of different scenarios in terms of correlations and the probability of joint extremes.

In a more applied approach, Cleland et al. study a strategy to be followed by a large electricity consumer to try to reduce prices by optimizing his/her consumption and reserve offers. Their results are based on the New Zealand electricity market. Luckny Zéphyr won the best student paper award at the conference with his work on a simplicial partitioning of the state space and its direct application to multi-reservoir systems. Concluding the research on Energy are Benedetto et al. with a new computational method to evaluate the predictability of energy markets based on the entropy levels of the time series.

The second part of the book is composed by contributions on Logistics in a wide sense. Those contributions range from conceptual problems to applications in logistics. A first set of contributions presents solving approaches to generic problems in distribution networks. Queiroz et al. and Macedo et al. both cover problems with location as well as routing decisions. In the former, that problem is considered in a scenario of uncertain demand, whereas the latter deals with the multi-trip variant of the problem. To conclude this set of works on generic problems, Huart et al. propose a heuristic for the time-dependent vehicle routing problem with time windows.

A second set of contributions studies logistics networks. Mazalov et al. study a non-cooperative game based approach to a passenger's transportation network. On the other hand, DeVos and Raa study the alternatives of collaboration in a decentralized distribution network.

Finally, there is a third set of contributions that includes more practical problems related to logistics. Heshmati et al. study a problem of scheduling cranes in a freight railway terminal, and Oliveira et al. investigate the problem of vehicle reservation, and the associated scheduling issue, and propose a GRASP algorithm.

The third part focuses on Production with topics ranging from scheduling and planning to supply chain disruption and impact of human factor. Lean manufacturing systems have put a significant amount of emphasis on cell production in terms of product grouping, clustering, tool selection and process flows. Dong and Hao, on the other hand, direct their attention towards the impact of the human factor in cell production and aim at capturing the operator's aptitude using a series of questionnaires. Martins suggests a mixed integer linear programming formulation to simultaneously model the firm's most important functional aspects: production, workforce and financial planning. Cardoso et al. study the impact on the supply chain from several disruptions together with demand uncertainty. Their findings are applied to a real case in Europe, with a production plant in Bilbao, and suppliers spread not only throughout Spain, but also in other european countries.

Also concerned with supply chains, but with the fair transfer of profit between all participants in the supply chain, Liu et al. propose a mixed integer linear programming model for the production and planning of global supply chains. The aim is therefore to determine the optimal transfer prices of products between plants and markets. From a different perspective, Mota et al. formulate a model concerned not only with supply chain planning, but especially with the incorporation of corporate social responsibility in these decisions, as people, firms and government are becoming more aware of all the issues with regional development.

Production planning and scheduling is also a hot topic. Braga et al. present a combined approach of the cutting and packing problem with scheduling, with an exact and compact assignment formulation. Virgílio et al., on the other hand, focus on the mould industry, developing an integer linear programming model based on discrete make-to-order job shop production. Also with a very practical case, Ospina et al. present the application of a mixed integer model for production planning in a Portuguese roasted coffee company. The production plan starts at warehousing and continues through to blending, roasting, grinding, packaging and warehousing again of the final product.

Finally, Rocha et al. analyse, in the context of a nesting problem, the trade-off between aggregating constraints and the subsequent reduction in computational costs, and the quality of the final solution. Nesting problems are very common in large industries where the need to allocate space and/or place sets of piece arises.

The fourth, and last, part of the book is devoted to optimization methods. In this part, one can find different approaches to distinct problems. A group of contributions proposes column generation approaches. Albornoz and Ñanco use a column generation procedure to produce optimum values for a zone delineation problem. Barbosa et al. suggest column generation based metaheuristics for the bus driver rostering problem. Alvelos et al. propose a matheuristic based on column generation to deal with a machine scheduling problem.

Some other combinatorial optimization problems are studied in two contributions. Vilà and Pereira present a genetic algorithm approach to a bin-packing problem. Leasege and Poss suggest a dynamic programming based procedure for the partial choice recoverable knapsack problem.

There are also two contributions dealing with stochastic problems. Kulikov and Gusyatnikov study a stochastic optimization problem associated with the decision on the stopping times for fractional Brownian motion. Hochreiter provides a simplification scheme for multi-stage optimization under uncertainty.

Finally, on a different level, Nicola et al. present a new scheme for assessing and quantifying value for the customer by using a fuzzy AHP method, and Bauso and Norman study population games and provide a new perspective on approachability.

Lisbon, Portugal
May 2014

Raquel J. Fonseca
João Telhada
Gerhard-Wilhelm Weber

Acknowledgements

We would like to thank all of the following reviewers for their contribution, assuring scientific quality of this volume:

- Bernardo Almada-Lobo, Faculty of Engineering of Porto University, Portugal
- Teresa Alpuim, Faculdade de Ciências da Universidade de Lisboa, Portugal
- Filipe Alvelos, Universidade do Minho, Portugal
- Cláudio Alves, Universidade do Minho, Portugal
- Ana Paula Barbosa-Póvoa, Instituto Superior Técnico, Portugal
- Paula Bouzas, University of Granada, Spain
- Maria Antónia Carravilla, INESC TEC, Faculdade de Engenharia, Universidade do Porto, Portugal
- Marta Castilho Gomes, Instituto Superior Técnico, Universidade de Lisboa, Portugal
- Giorgio Consigli, University of Bergamo, Italy
- Ana Luisa Custódio, Universidade Nova de Lisboa, Portugal
- Christophe Duhamel, LIMOS, Université Blaise Pascal, Clermont-Ferrand, France, France
- Jonas Egerer, TU Berlin, Germany
- Laureano Escudero, Universidad Rey Juan Carlos, España
- Paolo Foschi, University of Bologna, Italy
- Cristian Gatu, "Alexandru Ioan Cuza" University of Iasi, Romania
- Maria Isabel Gomes, Centro de Matemática e Aplicações - FCT, Universidade Nova de Lisboa, Portugal
- Nalan Gulpinar, Warwick Business School, United Kingdom
- Verena Hagspiel, Norwegian University of Science and Technology, Norway
- Grani Hanasusanto, Imperial College London, United Kingdom
- Alain Haurie, University of Geneva, Switzerland
- Carlos Henggeler Antunes, University of Coimbra, Portugal
- Ronald Hochreiter, WU Vienna University Economics and Business, Austria
- Dan Iancu, Stanford University, United States
- Daniel Kuhn, École Polytechnique Fédéral de Lausanne, Switzerland

- Songsong Liu, University College London, United Kingdom
- João Carlos Lourenço, Universidade de Lisboa, Instituto Superior Técnico, CEG-IST, Portugal
- Inês Marques, Faculdade de Ciências, Universidade de Lisboa - Centro de Investigação Operacional, Portugal
- Pedro Martins, Polytechnic Institute of Coimbra and Centro de Investigação Operacional, Portugal
- Martim Moniz, Université Libre de Bruxelles, Belgium
- Vittorio Moriggia, University of Bergamo, Italy
- José Fernando Oliveira, INESC TEC, Faculty of Engineering, University of Porto, Portugal
- Florentina Paraschiv, University of St. Gallen, Switzerland
- Andrew J. Parkes, University of Nottingham, United Kingdom
- Panos Parpas, Imperial College London, United Kingdom
- João Patrício, Instituto Politécnico de Tomar and Instituto de Telecomunicações, Portugal
- Gloria Pérez, Basque Country University, Spain
- Mustafa Pinar, Bilkent University, Turkey
- Diogo Pinheiro, Brooklyn College of the City University of New York, United States
- Marcus Poggi, PUC-Rio, Brazil
- Michael Poss, CNRS, Université de Technologie de Compiegne, France
- Tomasz Radzik, King's College London, United Kingdom
- Ana Respício, Faculdade de Ciências, Universidade de Lisboa - Centro de Investigação Operacional, Portugal
- Celso Ribeiro, Universidade Federal Fluminense, Brazil
- Ruediger Schultz, University of Duisburg-Essen, Germany
- Manuel Scotto, University of Aveiro, Portugal
- Luis Seco, University of Toronto, Canada
- Raghu Sengupta, Indian Institute of Technology Kanpur, INDIA
- Afzal Siddiqui, University College London, United Kingdom
- Amaro de Sousa, Universidade de Aveiro, Portugal
- A. Ismael F. Vaz, University of Minho, Portugal
- Alex Weissensteiner, Technical University of Denmark, Denmark
- Rafał Weron, Wrocław University of Technology, Poland
- Wolfram Wiesemann, Imperial College Business School, United Kingdom
- David Wozabal, Technical University of Munich, Germany
- Patrícia Xufre, Nova School of Business and Economics, Universidade Nova de Lisboa, Portugal
- Georges Zaccour, GERAD, HEC Montréal, Canada
- Mateusz Zotkiewicz, Warsaw University of Technology, Poland

Contents

List of Contributors

Víctor M. Albornoz Departamento de Industrias, Universidad Técnica Federico Santa María, Santiago, Chile

Filipe Alvelos Centro Algoritmi and Departamento de Produção e Sistemas, Universidade do Minho, Braga, Portugal

Cláudio Alves Universidade do Minho, Braga, Portugal

Marina Andretta Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Paulo, Brazil

Vítor Barbosa Escola Superior de Ciências Empreariais, Instituto Politécnico de Setúbal, Setúbal, Portugal

Ana Paula Barbosa-Póvoa CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal

Dario Bauso DICGIM, Università di Palermo, Palermo, Italy

Francesco Benedetto Department of Economics, "Roma Tre" University, Rome, Italy

Nuno Braga Universidade do Minho, Braga, Portugal

Gilles Caporossi GERAD and HEC Montréal, Montréal, QC, Canada

Sónia R. Cardoso CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal

Maria Antónia Carravilla INESC TEC, Faculty of Engineering, University of Porto, Porto, Portugal

Ana Carvalho CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal

Margarida Catalão-Lopes CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal

Pascal Côté Rio Tinto Alcan, Énergie électrique, Jonquière, QC, Canada

Nigel Cleland University of Auckland, Auckland, New Zealand

Thiago Alves de Queiroz INESC TEC, Faculty of Engineering, University of Porto, Porto, Portugal

Benedikt De Vos Ghent University, Zwijnaarde, Belgium

Yanwen Dong Cluster of Science and Technology, Fukushima University, Fukushima, Japan

Christophe Duhamel LIMOS, Université Blaise Pascal, Clermont-Ferrand, France

Eduarda Pinto Ferreira Porto Polytechnic-School of Engineering, GECAD, Porto, Portugal

J. J. Pinto Ferreira Universidade do Porto, Faculdade de Engenharia/DEIG, INESC TEC/UITT, Porto, Portugal

Christoph M. Flath Institute of Information Systems and Marketing (IISM), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

Roberto Fucarino Dipartimento di Ingegneria Chimica, Gestionale, Informatica, Meccanica, Università degli Studi di Palermo, Palermo, Italy

Johannes Gärttner Information Process Engineering (IPE), FZI Research Center for Information Technology, Karlsruhe, Germany

Gaetano Giunta Department of Engineering, "Roma Tre" University, Rome, Italy

A. Miguel Gomes INESC-TEC, Faculdade de Engenharia, Universidade do Porto, Porto, Portugal

Maria Isabel Gomes Centro de Matematica e Aplicações, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Caparica, Portugal

Marta Castilho Gomes CESUR, Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal

Pavel P. Gusyatnikov Moscow Institute of Physics and Technology, Moscow, Russia

Saïd Hanafi Université de Valenciennes, Valenciennes, France

Xiying Hao Cluster of Science and Technology, Fukushima University, Fukushima, Japan

Sam Heshmati INESC TEC, Faculty of Engineering, University of Porto, Porto, Portugal

Ronald Hochreiter Department of Finance, Accounting and Statistics, WU Vienna University of Economics and Business, Vienna, Austria

Vincent Huart GERAD and HEC Montréal, Montréal, QC, Canada

Zafeiris Kokkinogenis Artificial Intelligence and Computer Science Laboratory, Faculty of Engineering, University of Porto, Porto, Portugal

Alexander V. Kulikov Moscow Institute of Physics and Technology, Moscow, Russia

Bernard F. Lamond Operations and Decision Systems Department, Université Laval, Québec, QC, Canada

Pascal Lang Operations and Decision Systems Department, Université Laval, Québec, QC, Canada

Clément Lesaege UMR CNRS 7253 Heudiasyc, Université de Technologie de Compiègne, Centre de Recherches de Royallieu, Compiègne, France

Songsong Liu Centre for Process Systems Engineering, Department of Chemical Engineering, UCL (University College London), London, UK

Henrique Lopes Centro Algoritmi, Universidade do Minho, Braga, Portugal

Manuel Lopes CIDEM-ISEP, School of Engineering, Polytechnic of Porto, Porto, Portugal

João Carlos Lourenço CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal

Rita Macedo Université de Valenciennes, Valenciennes, France

Pedro Martins Polytechnic Institute of Coimbra, ISCAC, Coimbra, Portugal

Operations Research Center (CIO), University of Lisbon, Lisbon, Portugal

Loretta Mastroeni Department of Economics, "Roma Tre" University, Rome, Italy

Vladimir Mazalov Institute of Applied Mathematical Research, Petrozavodsk, Russia

Flávio Keidi Miyazawa Institute of Computing, State University of Campinas, Campinas, Brazil

Nenad Mladenović Université de Valenciennes, Valenciennes, France

Bruna Mota CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal

Pierre-Antoine Mudry Institute for Operations Research and Computational Finance, University of St. Gallen, St. Gallen, Switzerland

Linco J. Ñanco Departamento de Industrias, Universidad Técnica Federico Santa María, Santiago, Chile

Susana Nicola Porto Polytechnic-School of Engineering, GECAD, Porto, Portugal

Thomas W. L. Norman Magdalen College, Oxford, UK

Augusto Q. Novais LNEG, Laboratório Nacional de Energia e Geologia, Lisboa, Portugal

Beatriz Brito Oliveira INESC TEC, Faculty of Engineering, University of Porto, Porto, Portugal

José Fernando Oliveira INESC TEC, Faculty of Engineering, University of Porto, Porto, Portugal

Diana Yomali Ospina Department of Production and Mechanical Engineering, Universidad Autónoma de Manizales, Manizales, Colombia

Lazaros G. Papageorgiou Centre for Process Systems Engineering, Department of Chemical Engineering, UCL (University College London), London, UK

Florentina Paraschiv Institute for Operations Research and Computational Finance, University of St. Gallen, St. Gallen, Switzerland

Jordi Pereira Departamento de Ingeniería Industrial, Universidad Católica del Norte, Antofagasta, Chile

Sylvain Perron GERAD and HEC Montréal, Montréal, QC, Canada

Michael Poss UMR CNRS 5506 LIRMM, Université de Montpellier, Montpellier Cedex, France

Geoff Pritchard University of Auckland, Auckland, New Zealand

António Quintino CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal

Birger Raa Ghent University, Zwijnaarde, Belgium

Bruna Ramos Universidade do Minho, Braga, Portugal

Susana Relvas CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, Lisboa, Portugal

Ana Respício Centro de Matemática, Aplicações Fundamentais e Investigação Operacional/Departamento de Informática, FCUL, Universidade de Lisboa, Lisboa, Portugal

Pedro Rocha INESC-TEC, Faculdade de Engenharia, Universidade do Porto, Porto, Portugal

Rui Rodrigues INESC-TEC, Faculdade de Engenharia, Universidade do Porto, Porto, Portugal

Rosaldo J.F. Rossetti Artificial Intelligence and Computer Science Laboratory, Faculty of Engineering, University of Porto, Porto, Portugal

Anna Shchiptsova Institute of Applied Mathematical Research, Petrozavodsk, Russia

Yulia Tokareva Transbaikal State University, Chita, Russia

Franklina M.B. Toledo Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Paulo, Brazil

José Valério de Carvalho Universidade do Minho, Braga, Portugal

Mariona Vilà Escola Universitària d'Enginyeria Tècnica Industrial, Universitat Politècnica de Catalunya, Barcelona, Spain

Bárbara Esperança Virgílio GECO, Rua da Arroteia, Telheiro, Portugal

Christof Weinhardt Institute of Information Systems and Marketing (IISM), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany

Brent Young University of Auckland, Auckland, New Zealand

Golbon Zakeri University of Auckland, Auckland, New Zealand

Luckny Zéphyr Operations and Decision Systems Department, Université Laval, Québec, QC, Canada

Part I
Energy

Evaluating Price Risk Mitigation Strategies for an Oil and Gas Company

António Quintino, João Carlos Lourenço, and Margarida Catalão-Lopes

1 Introduction

Oil and gas (O&G) companies' earnings are substantially affected by the price fluctuations of crude oil, natural gas and refined products, which lead these companies to find ways to minimize their exposure to price risk. The work on investments and selection of efficient portfolios [9], along with the deregulation of energy markets in the United States in the 1980s, exponentiated the derivatives use in energy trade, to reduce companies' price risk exposures [1]. This research intends to evaluate the differences between hedging at business units (BU) level and hedging at company level, assuming that the risk tolerance at company level inherits the logic of the BU risk attitudes, through the "theory of syndicates" [15]. In this paper we use as case study a European O&G company, which manages its price risk separately at each BU. The axioms for utility as a decision criterion defined by von Neumann and Morgenstern [14] assured solid ground for the relation between financial measures, utility functions and corporate risk tolerance [6].

The remainder of this paper is organized as follows. Section 2 describes relevant measures and methods, Sect. 3 presents the results, Sect. 4 discusses them and presents the conclusions.

A. Quintino (✉) • J.C. Lourenço • M. Catalão-Lopes
CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal
e-mail: antonio.quintino@tecnico.ulisboa.pt; joao.lourenco@tecnico.ulisboa.pt; mcatalao@tecnico.ulisboa.pt

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_1

2 Relevant Measures and Methods

2.1 Earnings Formulation

The company is organized in three business units: the Exploration unit, the Refining unit and the Natural Gas unit. Since this research is focused on price risk, we take as reference the gross margin, calculated as the difference between the value of the goods bought and sold (crude oil, refined products and natural gas). The Exploration gross margin m_e is given by:

$$m_e = e_p \cdot p + e \,, \tag{1}$$

where e_p is the *Entitled Production* quantity in barrels of crude oil (bbl) for "Production Sharing Contracts" regimes, p is the crude price (\$/bbl) and e are the earnings (\$) in "Concession" regimes. The Refining gross margin m_r is given by:

$$m_r = \left(\sum_{i=1}^{n} y_i \cdot x_i - p \right) \cdot q_r \,, \tag{2}$$

where y_i is the yield (the oil industry name for the percentage of each i refined product taken from a unit of crude), x_i is the unitary price of each refined product i, p is the unitary price of crude and q_r is the yearly crude quantity refined (in tonnes). The Natural Gas gross margin m_g is given by:

$$m_g = \left(\sum_{i=1}^{n} z_i \cdot s_i - \sum_{j=1}^{k} w_j \cdot b_j \right) \cdot q_g \,, \tag{3}$$

where s_i and b_j are respectively the selling and buying price indexes, z_i and w_j are respectively the selling and buying yields, and q_g is the yearly total quantity of natural gas, measured in m^3 or kWh. As the goal underneath this research is to assume at least 1 year term hedging, we will choose the most traded derivatives in the OTC (over the counter) energy market: swap contracts. For each BU i, considering the yearly gross margin m_i, the yearly earnings e_i are given by:

$$e_i = m_i + \sum_{t=1}^{12} (f_i - s_{i_t}) \cdot q_i \,, \tag{4}$$

where f_i is the initial agreed fixed price for the swap, usually the average forward prices for the contract duration, s_{i_t} is the respective spot price at each future settlement month t and q_i is the swap notional quantity, having the same unit (bbl, weight, kWh) as the physical underlying item for each BU i.

2.2 Stochastic Prices Models and Risk Measures

Crude and refined products prices are modelled by their past monthly price returns [11].The historic price return r_t (in %) for crude or each refined product is:

$$r_t = \ln\left(\frac{p_t}{p_{t-1}}\right) , \tag{5}$$

where p_t is the average price in month t and p_{t-1} is the average price in month $t-1$. Each future stochastic price f_{t+1} under a GARCH(1, 1) process [4], with no-arbitrage and no-dividends assumptions, depends on the previous s_t price:

$$f_{t+1} = s_t \cdot \exp(r_t) , \tag{6}$$

where r_t is the stochastic price return, modelled by a combination of a GARCH(1, 1) process [4] and a t-copula function [13], the Copula-GARCH model [7]:

$$r_t = \left[\omega + \alpha \cdot r_{t-1}^2 + \beta \cdot \sigma_{t-1}^2\right]^{1/2} \cdot T_d\theta\left[t_d^{-1}(u_t)\right] \tag{7}$$

where ω is the constant term for variance, the conditional variance σ_{t-1}^2 assumes an autoregressive moving average process (ARMA), with α weighing the moving average part and β affecting the auto-regressive part, T is the t-copula with d degrees of freedom and correlation matrix θ, t^{-1} is the inverse Student's t distribution with d degrees of freedom and u_n are the variables' marginal distributions (the price returns residuals). The SIC-Schwarz and the AIC-Akaike information criteria were used as goodness of fit measures for GARCH and Copula [5], confirming the kurtosis excesses and fat tails characteristics in the referred prices' returns. Unlike the Gaussian copula, t-copulas preserve the tail dependence in extreme events. The Copula-GARCH method was implemented trough a multi-period Monte Carlo simulation [10] varying t from 1 to 12 months.

Conditional Value-at-Risk (CVaR) is a coherent risk measure [12], appraising how large is the average earnings (or losses) into the left and right distribution tails.

$$CVaR_{(\alpha)} = \frac{\int_a^b e \cdot f(e)\mathrm{d}e}{1-\alpha} \tag{8}$$

where $f(e)$ is the earning density function with F^{-1} being the inverse of the $f(e)$ cumulative distribution. For the left tail, $\alpha = 1\,\%$ (the level of significance assumed), $a = -\infty$, $b = F^{-1}(1\,\%)$. For the right tail, $\alpha = 99\,\%$, $a = F^{-1}(99\,\%)$, $b = +\infty$.

In fact, for all investors, swap contracts hedging is all about giving up potential upper gains (right tail) in exchange for having lower losses (left tail), so both extreme CVaR matter.

2.3 Risk Tolerance and Optimization

The selection of the optimal derivatives portfolio is influenced by the decision-maker's attitudes towards financial risk. The expected value of an utility function is the utility of the *certainty equivalent* (*CE*) [3] considering the exponential utility function as the most appropriate [8], we have:

$$CE \approx E(x) - \left(\frac{\sigma^2(x)}{2\rho}\right), \quad (9)$$

where x is the stochastic earnings variable, $E(x)$ is the earnings expected value, $\sigma^2(x)$ is the earnings variance and ρ is the risk tolerance, evaluated trough one questionnaire assessment for each BU. The selection of the optimal derivatives portfolio is achieved by stochastic optimization [10], having the swap notional quantity q_i in (4) as the decision variable "inside" each BU earning e_i.

$$\text{Max } CE_i \approx \text{Max } \left(E(e_i) - \frac{\sigma^2(e_i)}{2\rho_i}\right) \quad (10)$$

After obtaining the optimal derivatives portfolio for each BU i, we enter each i solution in an additive corporate hedging simulation, which we named Program 1.

In a second approach, named Program 2, we consider the company's earnings e_c given by $e_c = e_e + e_r + e_g$, where e_e, e_r, e_g are respectively the exploration, refining and natural gas earnings. According to the Theory of Syndicates [15] the risk tolerance of the company (ρ_c) can be assumed to be the sum of each BU risk tolerance ρ_i. Therefore, replacing e_i by e_c and ρ_i by ρ_c in expression (10) will allow us to maximize, assuming exponential utility functions for each BU. The Certainty Equivalent for the whole company.

3 Results

The results from Monte Carlo simulation before hedging are presented in Table 1. Tables 2 and 3 present the results of Program 1 and Program 2, respectively. The "*% of Hedge*" solution for each BU is the ratio between the notional amounts of swap contracts q_i and the respective BU yearly physical production.

Table 1 Results before hedging (in $ million)

Measures	Refining	Natural gas	Exploration	Company
$E(e_i)$	273	124	424	821
$\sigma(e_i)$	102	3	12	98
$CVaR_{99\,\%}(e_i)$	665	113	469	1220
$CVaR_{1\,\%}(e_i)$	−87	116	393	488

Table 2 Program 1, results after BU hedging (in $ million)

Measures	Refining	Natural gas	Exploration	Company
$E(e_i)$	268	125	423	815
$\sigma(e_i)$	33	0	2	33
$CVaR_{99\,\%}\,(e_i)$	436	125	431	985
$CVaR_{1\,\%}\,(e_i)$	161	125	418	713
ρ	173	3	22	198
CE	264	125	423	813
% of Hedge solution	79	100	35	81

Table 3 Program 2, results after Company hedging (in $ million)

Measures	Refining	Natural gas	Exploration	Company
$E(e_i)$	269	125	424	817
$\sigma(e_i)$	37	0	11	35
$CVaR_{99\%}\,(e_i)$	447	125	461	1003
$CVaR_{1\%}\,(e_i)$	153	125	395	722
ρ	173	3	22	198
CE	265	125	421	814
% of Hedge solution	69	100	0	72

4 Discussion and Conclusions

The results in Table 2 present a significant decrease in the company's earnings uncertainty from the initial unhedged situation in Table 1. The earnings standard deviation reduces sharply from $ 98 million to $ 33 million and the minimum gains, measured by $CVaR_{1\,\%}$, increase from $ 488 million to $ 713 million at the cost of the maximum gains, measured by $CVaR_{99\,\%}$, shrinking from $ 1220 million to $ 985 million.

Comparing the results of Table 2 with Table 3, we observe that the certainty equivalent and the standard deviation have negligible changes. However, in Table 3 the extreme tails shows both higher gains and the optimal solution (i.e. the *% of Hedge*) reduces from 81 to 72 %, implying less payout exposure. Under Program 2, the Exploration crude price risk is absorbed (0 *% of Hedge*) by Refining, eliminating the risk overlapping and reducing the Refining hedge from 79 to 69 %.

We conclude that hedging at company level (Program 2) clearly outperforms the BU individual hedging (Program 1). Since the certainty equivalents of both programs are quite similar, we propose further research to include other criteria to evaluate the final hedging results. Multi-Attribute Value Theory [2] should be applied to assess decision-makers' preferences upon extreme tails and payout exposure changes.

References

1. Administration, E.I.: Derivatives and Risk Management in the Petroleum, Natural Gas and Electricity Industries. U.S. Department of Energy, Washington (2002)
2. Belton, V., Stewart, T.J.: Multiple Criteria Decision Analysis: An Integrated Approach. Springer, New York (2002)
3. Ben-Tal, A., Teboulle, M.: Expected utility penalty functions and duality in stochastic nonlinear programming. Manag. Sci. **32**(11), 1445–1446 (1986)
4. Bollerslev, T.: Generalized autoregressive conditional heteroskedasticity. J. Econ. **31**(3), 307–327 (1986)
5. Fermanian, J.D.: Goodness-of-fit tests for copulas. J. Multivar. Anal. **95**(1), 119–152 (2005)
6. Howard, R.A.: Decision analysis: practice and promise. Manag. Sci. **34**(6), 679–695 (1988)
7. Jondeau, E., Rockinger, M.: The copula-garch model of conditional dependencies: an international stock market application. J. Int. Money Financ. **25**(5), 827–853 (2006)
8. Kirkwood, C.W.: Approximating risk aversion in decision analysis applications. Decis. Anal. **1**(1), 51–67 (2004)
9. Markowitz, H.: Portfolio Selection: Efficient Diversification of Investments. Wiley, New York (1959)
10. ModelRisk: Vose-sofware. http://www.vosesoftware.com/ (2012)
11. Platts: Market data - oil. http://www.platts.com/ (2012)
12. Rockafellar, R.T., Uryasev, S.: Optimization of conditional value-at-risk. J. Risk **2**(3), 21–41 (2000)
13. Sklar, A.: Fonctions de répartition à n dimensions et leurs marges. Publ. Inst. Stat. Univ. Paris **8**, 229–231 (1959)
14. von Neumann, J., Morgenstern, O.: Theory of Games and Economic Behavior. Princeton University Press, Princeton (1947)
15. Wilson, R.: The theory of syndicates. Econometrica **36**(1), 119–132 (1968)

Load Shifting, Interrupting or Both? Customer Portfolio Composition in Demand Side Management

Johannes Gärttner, Christoph M. Flath, and Christof Weinhardt

1 Introduction

In recent years, power grids across the globe have seen a dramatic increase in variable generation assets [5]. At the same time, integration of these resources has not been actively addressed. Subramanian et al. [8] note that current approaches work with "today's modest penetration levels, but will not scale [effectively] tomorrow". A more cost-efficient integration of variable energy sources calls for a more flexible demand side. This will allow to limit the required expensive balancing and storage capacities [5]. Smart Grid systems and novel incentive schemes will play a key role to achieve this goal. Smart Grids enhance the existing grid infrastructure through the provision of bi-directional information and communication technology. In this context, the activation of the historically passive demand through demand side management (DSM) is a central theme. In general, it is expected that the power system will have to change from a system of flexible generation serving random loads to a system of flexible loads adjusting to fluctuating generation. Consequently, system operators will in the future be less concerned about handling demand uncertainty, but rather need to focus on supply uncertainty.

Prior research has established the balancing potentials offered by scheduling flexible load portfolios [8]. Apart from that, little is known with respect to forming

J. Gärttner (✉)
Information Process Engineering (IPE), FZI Research Center for Information Technology, Haid-und-Neu Str. 10-14, 76131 Karlsruhe, Germany
e-mail: gaerttner@fzi.de

C.M. Flath • C. Weinhardt
Institute of Information Systems and Marketing (IISM), Karlsruhe Institute of Technology (KIT), Englerstraße 14, 76131 Karlsruhe, Germany
e-mail: flath@kit.edu; weinhardt@kit.edu

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_2

these portfolios in the first place. Thus, it is necessary to develop methods that guide the design of demand response portfolios. We present an optimization model to guide the formation and subsequent scheduling of demand response portfolios for energy retailers. We illustrate our model using an example scenario based on empirical load and solar generation data.

2 Related Work

DSM, i.e. the active coordination of load, can offer sizeable control potentials at much lower costs than the expansion of storage capacities [7]. To maximize the benefits obtained from contracted flexible loads, operators need to optimally despatch these loads. Parvania and Fotuhi-Firuzabad [6] schedule load shifting and curtailment as well as decentral generation assets to minimize wholesale electricity costs. Using different scheduling routines, Subramanian et al. [9] show that efficient DSM can already be achieved with modest load flexibility endowments.

Besides this scheduling-oriented literature, demand response assets have also been evaluated with respect to portfolio design concerns. Abstracting from individual load dispatching, this stream of literature analyzes generic demand entities to identify efficient portfolio composition rules. Baldick et al. [1] determine the value and optimal execution of demand-interruption programs using option-pricing techniques. Deng and Xu [3] also consider interruptible load contracts and propose a mean-risk analysis to guide the portfolio design decision. Valero et al. [10] use data mining techniques to test customer demand and response options in different price scenarios.

Our work tries to connect these branches of literature by accounting for the utilization of customer-level load characteristics while at the same time accounting for prior portfolio-design activities.

3 Scenario and Optimization Model

Load scheduling decisions determine which electrical loads are served at what time. The attainable scheduling quality (with respect to a given objective) critically hinges on the composition of the underlying customer portfolio. The customer portfolio design decision needs to determine which loads to contract as well as the corresponding contracting terms. We reflect the interdependency between portfolio design and load scheduling as a two-stage problem. In the first stage, an electricity retailer determines the composition of the customer portfolio. The second stage handles the optimal load scheduling of the chosen portfolio. The main challenge is the integration of fluctuating and hence uncertain renewable generation.

In this paper, we look at the offline integrated optimization problem that simultaneously determines customer selection and load scheduling as a deterministic benchmark. We consider a set C of customers over a horizon of T time slots. Customers are indexed $c = 1, \ldots, |C|$, time slots $t = 1, \ldots, |T|$. Customer demand is assumed to consist of three components:

- base load ($D^B_{c,t} \in \mathbb{R}_+$) must not be influenced and has to be satisfied at any time
- shiftable load ($D^S_{c,t} \in \mathbb{R}_+$) can be shifted over time, although a customer's total shiftable demand $\sum_{t \in T} D^S_{c,t}$ has to be fully covered over the optimization horizon
- interruptible load ($D^I_{c,t} \in \mathbb{R}_+$) can be served at any level from the interval $[I^P D^I_{c,t}, D^I_{c,t}]$ at any point in time ($I^P \in [0, 1]$: interrupting potential).

Fluctuating renewable energy supply is given by $R_t \in \mathbb{R}_+$. If demand deviates from supply, the deviation has to be balanced by conventional power. The variable costs of balancing power are given by $c^G \in \mathbb{R}_+$. We assume the resulting balancing power costs $C^G \in \mathbb{R}_0^+$ to be a quadratic function to reflect increasing marginal costs c^G.

Consumer portfolio design needs to decide which customers are contracted. Therefore, a subset C^C of the set of customers C is selected to be part of the portfolio and split in to three subsets: inflexible customers C^B, shiftable customers C^S, interruptible customers C^I. Note that the latter sets are not disjoint as a customer can be contracted to offer both his shiftable and interruptible load component. The scheduling variables determine for each customer how much load is shifted from one time slot to another ($X^S_{c,t,s} \in \mathbb{R}_0^+$) and how much interruptible load is served ($X^I_{c,t} \in [I^P D^I_{c,t}, D^I_{c,t}]$). Obviously, only customer loads contracted with shifting or interruption provisions can be controlled in this fashion with non-contracted loads being merged into the base load component.

When choosing a contract that allows load shifting or interrupting, a customer cedes control to the operator and needs to be compensated with more favorable electricity rates. Let p be the retail price of base load consumption. The price of shiftable demand is then obtained by applying a discount $\delta^S \in [0, 1]$ on the price of base load, that is $\delta^S p$. Whenever load shifting is triggered by the operator, a second cost component arises. These additional dispatch-related shifting costs depend on the load shifting distance which is a deviation-measure between a customer's original and the realized load schedule. The cost structure of interruptible load is fairly similar. A discount factor $\delta^I \in [0, 1]$ defines the price for interruptible load in relation to base load. Obviously, whenever load is interrupted, an electricity retailer forgoes any revenues from this load type. The concrete optimization model is presented in the Appendix.

In the following section we present preliminary results that show the influence of shifting—and interrupting discounts on the customer portfolio composition.

4 Evaluation

To evaluate our optimization model we use two different data sources. Demand data was retrieved from the Irish Social Science Data Archive.[1] This data set provides smart meter readings from over 5000 Irish homes and businesses and reports each customer's energy consumption in 30 min intervals. As this is aggregate load, there is no detailed information on the underlying load flexibility. To extract more information from aggregate load data collections, Carpaneto and Chicco [2] suggest interpreting residential load curve collections as probability distributions. Building upon this assessment, Flath [4] suggests approximating the underlying flexibility level using the likelihood of a certain demand level. Following this approach, we derive the demand components of a given customer by splitting up the collection of smart meter readings by the 30, 60 and 85 % quantile. We cut the full demand to smoothen outliers.

To model the variations of renewable generation, we use empirical solar generation data. We normalize the solar feed-in data to an appropriate scale of the demand level to ensure meaningful results.

To illustrate the dependencies of the portfolio composition and the optimal value from shifting—and interrupting discounts we chose the following scenario. Customers can be precluded from the portfolio. Customers that are part of the portfolio can be assigned to the following flexibility types: Base load, Shiftable, Interruptible and Flexible (a flexible customer offers both, shiftable and interruptible load). In order to minimize the influence of extreme demand and supply scenarios, we solved the integrated optimization problem ten times for each combination of δ^S and δ^I. Figure 1 shows the average share customers assigned to one of the flexibility types for given δ^S and δ^I.

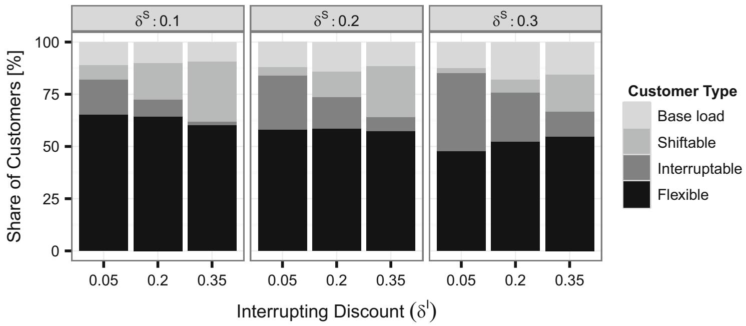


Fig. 1 Sensitivities of Customer Portfolio Composition by flexibility discounts

[1] http://www.ucd.ie/issda/data/commissionforenergyregulationcer/.

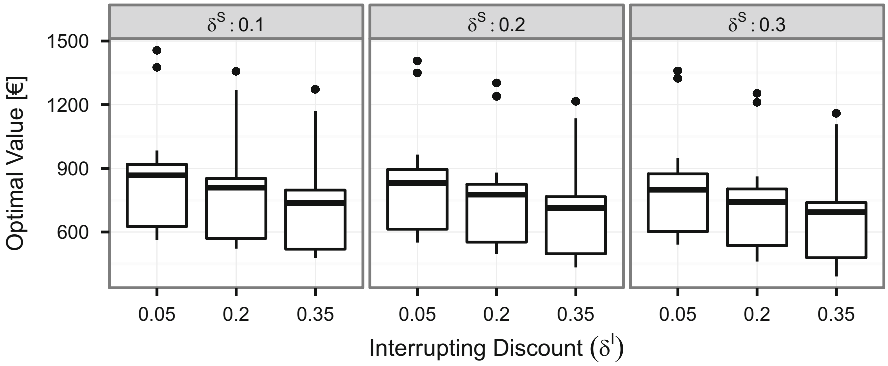


Fig. 2 Sensitivities of the optimal value by flexibility discounts

Surprisingly, there are no customers precluded from the portfolio in this scenario instance. This is caused by the normalization of supply and aggregate demand. Obviously, it becomes less attractive for an electricity retailer to pursue interruptible or shiftable contracts when the corresponding discount levels increase. This is because, *ceteris paribus*, the usage of balancing power will become more attractive than contracting flexibility at a high revenue loss. Both, the impact of changing discounts on base load—and flexible customers and the influence of shifting discounts seems rather minor. The effect of increasing interrupting discounts, on the other hand, is stronger. The share of interruptible contracts shrinks and is substituted by shiftable contracts.

Not surprisingly, the electricity retailer's overall profit (the optimal value, respectively) decreases with growing discounts (Fig. 2). Similar to the portfolio composition, the shifting discount influences the optimal value less than the interrupting discount. This results from the tariff design, that ensures that all shiftable load has to be served while interruptible load can be shedded. Therefore, when contracting interruptible load, not only a discount on served load has to be granted but there are no revenues from interrupted load at all.

5 Conclusion and Outlook

Designing optimal customer portfolios includes deciding on which customers should be part of the portfolio as well as how much and which type of flexibility each customer offers. In this paper we introduce a two-stage characterization of a stochastic optimization problem to select an optimal customer portfolio referring to subsequent load scheduling. As a benchmark we introduce an integrated optimization model to illustrate the influence of price discounts on the portfolio composition using empirical input data. Finally, we present exemplary results that clarify the impact on both, the customer portfolio composition and profits attainable.

For further research we plan to investigate if the shiftable and interruptible contracts behave like complements or substitutes. We will evaluate the model for both, stochastic demand and supply and to compare the results with the benchmark solution. Finally, our future research aims to better understand the effect of customer heterogeneity with respect to flexibility assets as well as contracting specifications.

Appendix

With the parameters and decision variables described in Sect. 3 the portfolio design problem is formulated as follows. First, we describe the model constraints. We then define the objective function and its components. Constraint (1) ensures that each customer's overall shiftable demand is covered over the optimization horizon.

$$\sum_{t\in T} D^S_{c,t} = \sum_{t\in T}\sum_{s\in T} X^S_{c,t,s} \quad \forall c \in C \tag{1}$$

Load shifted from t to s cannot exceed the gross shiftable load in t.

$$D^S_{c,t} \geq \sum_{s\in T} X^S_{c,t,s} \quad \forall c \in C, \forall t \in T \tag{2}$$

Similarly, dispatched interruptible load in t $X^I_{c,t}$ is bounded by the minimum dispatch amount and the gross interruptible load in t.

$$I^P D^I_{c,t} \leq X^I_{c,t} \leq D^I_{c,t} \quad \forall c \in C, \forall t \in T \tag{3}$$

Total load must equal the sum of available renewable supply and dispatched conventional generation:

$$G_t = \sum_{c\in C}\left(D^B_{c,t} + X^I_{c,t} + \sum_{s\in T} X^S_{c,t,s}\right) - R_t \quad \forall t \in T \tag{4}$$

The supplier's objective is to maximize profits which is given by revenues minus costs. We split the costs into two components, contracting costs and dispatching costs:

$$\max_{C^B,C^S,C^I,X^S,X^I} \underbrace{\sum_{t\in T}\sum_{c\in C} p\left(D^B_{c,t} + D^S_{c,t} + D^I_{c,t}\right)}_{\text{revenues}} - \underbrace{F^S + F^I}_{\text{contracting costs}} - \underbrace{C^G + C^S}_{\text{dispatching costs}} \tag{5}$$

Contracting costs occur during the portfolio design phase and are driven by the discounts on the two flexibility types:

$$F^S = \sum_{t \in T} \sum_{c \in C^S} p\delta^S D^S_{c,t} \qquad F^I = \sum_{t \in T} \left(\sum_{c \in C^I} p(D^I_{c,t} - (1-\delta^I)X^I_{c,t}) \right) \tag{6}$$

Dispatching costs reflect the usage of costly conventional generation and shifting distance penalties from shifting execution:

$$C^G = c^G \sum_{t \in T} (G_t)^2 \qquad C^S = c^S \sum_{c \in C} \sum_{t \in T} \Big(\underbrace{\sum_{s=0}^{t-1} (t-s)^2 X^S_{c,t,s}}_{\text{loads shifted forward}} + \underbrace{\sum_{s=t+1}^{T} (s-t)^2 X^S_{c,t,s}}_{\text{loads shifted backward}} \Big) \tag{7}$$

References

1. Baldick, R., Kolos, S., Tompaidis, S.: Interruptible electricity contracts from an electricity retailer's point of view: valuation and optimal interruption. Oper. Res. **54**(4), 627–642 (2006)
2. Carpaneto, E., Chicco, G.: Probabilistic characterisation of the aggregated residential load patterns. Gener. Transm. Distrib. IET **2**(3), 373–382 (2008)
3. Deng, S.J., Xu, L.: Mean-risk efficient portfolio analysis of demand response and supply resources. Energy **34**(10), 1523–1529 (2009)
4. Flath, C.M.: Flexible demand in smart grids-modeling and coordination. Ph.D. thesis, Karlsruhe, Karlsruher Institut für Technologie (KIT) (2013)
5. IEA: The power of transformation - wind, sun and the economics of flexible power systems. International Energy Agency, Paris (2014)
6. Parvania, M., Fotuhi-Firuzabad, M.: Demand response scheduling by stochastic scuc. IEEE Trans. Smart Grid **1**(1), 89–98 (2010)
7. Strbac, G.: Demand side management: benefits and challenges. Energy Policy **36**(12), 4419–4426 (2008)
8. Subramanian, A., Garcia, M., Dominguez-Garcia, A., Callaway, D., Poolla, K., Varaiya, P.: Real-time scheduling of deferrable electric loads. In: American Control Conference (ACC), 2012, pp. 3643–3650. IEEE (2012)
9. Subramanian, A., Garcia, M.J., Callaway, D.S., Poolla, K., Varaiya, P.: Real-time scheduling of distributed resources. IEEE Trans. Smart Grid **4**(4), 2122–2130 (2013)
10. Valero, S., Ortiz, M., Senabre, C., Alvarez, C., Franco, F., Gabaldon, A.: Methods for customer and demand response policies selection in new electricity markets. Gener. Transm. Distrib. IET **1**(1), 104–110 (2007)

Stress-Testing for Portfolios of Commodity Futures with Extreme Value Theory and Copula Functions

Pierre-Antoine Mudry and Florentina Paraschiv

1 Overview

The severity of the global financial crisis sparked a deep review of risk-management practices by regulators (see [1]). Basel III market risk framework requires banks to subject their portfolios to a series of simulated stress scenarios and to report the results to the supervisory authorities [1]. In particular, the post-crisis regulations criticize the overreliance on historical prices and correlations, and recommend instead a more rigorous analysis of extreme events.

One of the recent trends in the financial markets has been the increasing financialization of commodities, especially since the introduction of commodity indices [8]. The benefits for investors are manifold: it frees them from the risk of unwanted delivery, from the costs of storage, and from losses linked to the perishable nature of agricultural commodities, while allowing them to hedge against inflation, diversify their portfolios, and ride the boom triggered by the appetite of emerging nations for commodities. Given the exponential growth of investments in commodity indices by institutional investors, the question of adequately stress-testing those indices in the context of a broader portfolio of financial securities is of great interest. We apply a combined approach of extreme value theory (EVT) for modeling extreme movements in the risk factors and we look at the dependency structures in a dynamic way, with copula functions. To our knowledge, EVT and copulas have been extensively applied to equities and currencies portfolios (see [4, 5, 7]), but rarely to portfolios of commodity futures.

P.-A. Mudry • F. Paraschiv (✉)
Institute for Operations Research and Computational Finance, University of St. Gallen, Bodanstrasse 6, CH-9000 St. Gallen, Switzerland
e-mail: Pierre-Antoine.Mudry@gs.com; florentina.paraschiv@unisg.ch

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_3

2 Data and Methods

We analyze a portfolio of ten most important components in the DJ-UBS commodity index: WTI Light Sweet Crude Oil, Brent Crude Oil, Natural Gas, Corn, Wheat, Soybeans, Live Cattle, Gold, Aluminium, Copper. In total, these ten commodities represent 69 % of the total weights, making the chosen portfolio a good proxy for the index. In order to determine the weight of each commodity in the test portfolio, we simply divided each weight in the index by the combined weight of 69 %.

We used daily logarithmic returns, from 01 January 1998 to 31 December 2011. In the case of the reference index, the provider usually rolls the futures contracts over four times a year, depending on the most liquid contracts trading on a particular commodity. The provider therefore buys relatively short termed contract. For simplicity, it would have been very cumbersome to roll the contracts over in the same way as the index provider. We therefore limited ourselves to a bi-annual roll. Moreover, to soften the jumps linked to rolling over contracts, we used contracts with approximately 1 year maturity, that are rolled over 6 months before expiration.

To perform adequate stress-tests, a realistic model for the risk factors is required. A preliminary descriptive analysis of logarithmic returns shows patterns like: positive skewness, stationarity, autocorrelation and volatility clustering. We therefore model the returns with a AR-GARCH(1,1) asymmetric model. The lags were found by performing the Akaike (AIC) and Bayesian (BIC) information criteria.

$$y_t = \mu + \phi_1 y_{t-1} + \varepsilon_t \tag{1}$$

$$h_t^2 = a_0 + a_1 \varepsilon_{t-1}^2 + a_2 h_{t-1}^2 + b\psi(\varepsilon_{t-1})\varepsilon_{t-1}^2 \tag{2}$$

where h_t^2 is the conditional variance of ε_t, $z_t = \varepsilon_t / h_t$, with z_t $N(0,1)$ or Student's t-distributed (scaled to have variance 1) IID innovations with mean $= 0$, variance $= 1$, and degree of freedom parameter, ν. Additionally, an indicator function is introduced: $\psi(\varepsilon_{t-1}) = 1$ if ε_{t-1} (or z_{t-1}) is negative, or 0 if ε_{t-1} (or z_{t-1}) is positive. As there is no restriction on the sign of b, the model can be applied to describe both negatively or positively skewed data.

Two versions are tested: asymmetric AR-GARCH with normal and with t-innovations. A likelihood ratio test shows the superiority of the latter model version. This is not surprising, given the fat tails of commodity returns. We further produced a probability plot and compared the returns to the standard normal distribution and the fitted AR-GARCH(1,1) model with t-innovations. We observed that the model strongly underestimates extreme events. However, for a rigorous stress testing, exactly the extremely large returns are of importance. Embrechts et al. [3] and McNeil et al. [6] prove evidence for a good performance of a combined approach GARCH with parametric tails based on extreme value theory (EVT). For the center of the distribution, where most of the data are concentrated, kernel smooth interior is used for the estimation. However, for the tails, where usually data is scarce, a parametric approach based on extreme value theory is selected, whereas the generalized Pareto distribution is able to asymptotically describe the behavior of the tails. We will therefore apply this approach to model the standardized residuals z_t, in Eq. (2). The notation for a generalized Pareto (GP) distribution is introduced for any $\xi \in \mathbf{R}, \beta \in \mathbf{R}_+$ [6]:

$$GP_{\xi,\beta}(z) = 1 - \left(1 + \xi \frac{z}{\beta}\right)_+^{-\frac{1}{\xi}}, z \in \mathbf{R}$$

where item $1/\xi$ is known as the *tail index* and β a *scaling* parameter. The threshold u was fixed for 10 % uppermost and lowermost returns, for each commodity.

So far, we showed how we modeled the risk factors individually. However, for a realistic portfolio stress testing, the evolution of dependency structures among the considered commodities is of great importance. Given the contagion effect, it is expected and empirically observed that in times of market stress, joint extreme returns occur in commodity markets. We therefore model joint positive or negative returns with a t-copula. In the case of t-distributions the d-dimensional t-copula with ν degrees of freedom is given by:

$$C^t_{\nu,\Sigma}(u) = t_{\nu,\Sigma}(t_\nu^{-1}(u_1), \ldots, t_\nu^{-1}(u_d)) \tag{3}$$

where Σ is a correlation matrix, t_ν is the cumulative distribution function of the one dimensional t_ν distribution and $t_{\nu,\Sigma}$ is the cumulative distribution function of the multivariate $t_{\nu,\Sigma}$ distribution.

3 Estimation Results

Scenarios for multivariate stress tests can be constructed as historical, hybrid, or hypothetical scenarios [1]. While historical scenarios assume a repetition of past crises, in hybrid scenarios the historical market movements are only used to calibrate the process of risk factors evolution. Hypothetical scenarios are not restricted to a repetition of the past, but allow a more flexible formulation of potential events. In this study, we show the limitations of historical scenarios and the importance of a forward looking analysis in the context of hybrid scenarios.

The first stress test we consider is based on the derivation of historical scenarios. Creating scenarios with historical data is probably the most intuitive approach, since the events did happen in reality and are thus plausible to reappear. We construct the P&L of our portfolio for the next 22 days horizon starting at 1st January 2012, based on the returns of the risk factors empirically observed during the financial crisis from 28 March 2008 to 31 March 2010. In this case, we want to assess the portfolio losses in case of a repetition of a financial stress situation. The P&L of the portfolio under the simulated historical scenario is simply given by the empirical distribution of past gains and losses on this portfolio, during the financial crisis. The implementation of this non-parametric method is simple, since it does neither require a statistical estimation of the multivariate distribution of risk factor changes, nor an assumption of their dependence structure.

The second stress test to be considered is based on hybrid scenarios. The parameters and residuals of the AR-GARCH with EVT processes of the different commodity returns and the *t*-copula are calibrated on the financial crisis data ranging from 28 March 2008 to 31 March 2010. Based on these parameters, the risk factors are simulated for the next 22 days, 10,000 scenarios, and the P&L is finally constructed.

Figure 1 shows comparatively the P&L for the historical and the hybrid scenarios. The P&L for the historical scenario obviously displays the characteristic stepwise pattern. The more extreme the returns, the more the two distributions drift apart. However, the hybrid scenario overestimates positive returns significantly. One explanation for this lies in the symmetry of the t-copula, which struggles to account for skewed portfolio returns, despite its many merits [2]. The lower tail of the historical scenario distribution is truncated at $-32.93\,\%$, while the maximum simulated loss with the hybrid scenario is $-64.08\,\%$. Thus, for extreme tail quantiles, we observe that the historical scenario signals a much lower simulated loss with the hybrid scenario. This underlines the main drawbacks of the overreliance on the historical simulation method, as discussed in [1]: this method is unconditional, and it neglects the time-varying nature of financial time series, it neglects the dependence structure. Furthermore, based on a limited time span, extreme quantiles are difficult to estimate. This example shows additionally that the hybrid scenario is able to extrapolate beyond the historical data, which, from the view point of financial regulations [1], is a major feature of a realistic stress testing technique.

To show the importance of stress scenarios, we compare the P&L values derived from historical and hybrid scenarios with the P&L derived from the baseline

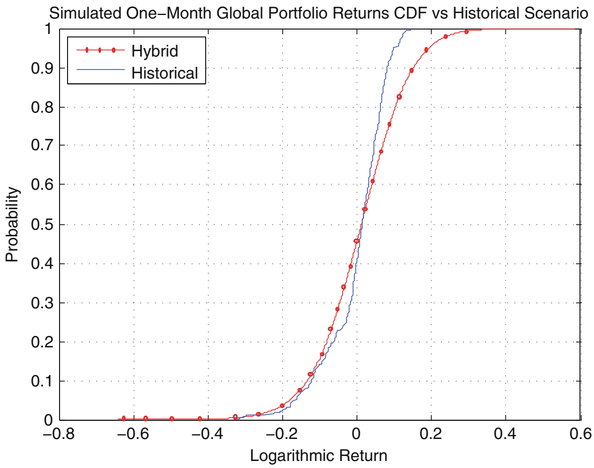


Fig. 1 Hybrid vs historical stress scenarios

Table 1 Metrics for hybrid and historical stress scenarios

Metric	Baseline	Hybrid	Historical
Degrees of freedom	12.79	15.25	N/A
Max. simulated loss	−37.86 %	−64.08 %	−32.93 %
Max. simulated gain	30.57 %	58.45 %	14.49 %
Simulated 90 % VaR	−4.67 %	−13.31 %	−12.19 %
Simulated 95 % VaR	−6.35 %	−17.96 %	−16.56 %
Simulated 99 % VaR	−9.64 %	−29.83 %	−28.31 %
Simulated 90 % ES	−6.92 %	−19.73 %	−18.06 %
Simulated 95 % ES	−8.40 %	−24.10 %	−21.86 %
Simulated 99 % ES	−11.58 %	−34.10 %	−31.17 %
Simulated 99.9 % ES	−16.93 %	−49.26 %	N/A
Simulated 99.99 % ES	−29.38 %	−63.78 %	N/A

scenarios. The latter aims at estimating the portfolio performance at the end of the 22 day period, without the impact of stress. We therefore calibrate the AR-GARCH model with EVT for the risk factors and the *t*-copula for interdependencies to the entire data sample: 01 January 1998 to 31 December 2011. With the simulations over 22 days for each risk factor, we recompute the P&L. Comparative statistics over risk measures are offered in Table 1. We observe that with the baseline scenarios, the risk of the portfolio, expressed by the VaR and Expected Shortfall (ES) for tail quantiles above 90 %, is significantly underestimated. Well identified stress scenarios are of great important for portfolio risk managers, as they quantify the magnitude of losses that might be expected in case of market stress.

In Table 1 we observe that the estimated degrees of freedom of the copula function for the hybrid scenario are higher than in the case of baseline scenario. This is surprising, since it indicates lower tendency of joint extremes in commodities during the financial crisis than in the overall investigated period. To better understand the cause for these results, we recalibrated our AR-GARCH with EVT and *t*-copula on a rolling time window, collected the degrees of freedom and additionally computed average correlations. We focused on 3-year rolling windows. The results are plotted in Fig. 2. Overall we observe that correlations among commodity returns increased, while the degrees of freedom show some oscillating patterns. Until 2006, we conclude that commodity returns became more correlated with each other, and joint extremes are more likely. However, during the boom and bust cycle of 2007–2009, and further during the 2008–2010 window, although correlations increased among commodities, we observe an increase in the degree of freedom as well. This confirms our previous results, that the tail dependence structures among commodities weakened during the financial crisis. A possible explanation for this are the different dynamics among commodity prices during the financial crisis: some underwent a relatively moderate growth and fall (agricultural commodities), while others, (oil, gas, copper) went through a massive boom and bust cycle.

Our results show that the reliance on standard assumptions like the increase in the probability of joint extremes among financial assets in times of market

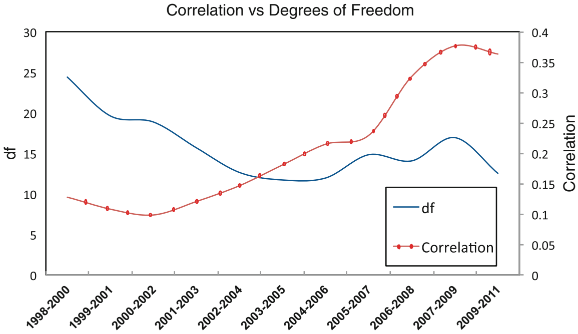


Fig. 2 Rank correlations vs degrees of freedom

stress is not always realistic. By contrary, we show that there have been structural breaks in commodity markets that temporarily led to a breakdown of expected statistical patterns, like tail dependence structures. This fact should be explored by risk managers in hypothetical scenarios, by shocking arbitrary combinations of market factors, volatilities, and dependence structures. The pure reliance on historical assumptions has serious limitations for stress testing.

References

1. Bank for International Settlements: Principles for Sound Stress Testing Practices and Supervision. Basel Committee on Banking Supervision, Basel (2009)
2. Demarta, S., McNeil, A.: The t copula and related copulas. Int. Stat. Rev. **73**(1), 111–129 (2005)
3. Embrechts, P., Klüppelberg, C., Mikosch, T.: Modeling extremal events: for insurance and finance, 4th printing edn. Springer, Berlin (2003)
4. Gencay, R., Selcuk, F.: Extreme value theory and value-at-risk: relative performance in emerging markets. Int. J. Forecast. **20**(2), 287–303 (2004)
5. Haile, F., Pozo, S.: Exchange rate regimes and currency crises: an evaluation using extreme value theory. Rev. Int. Econ. **14**(4), 554–570 (2006)
6. McNeil, A.J., Frey, R., Embrechts, P.: Quantitative Risk Management: Concepts, Techniques and Tools. Princeton University Press, Princeton (2005)
7. Patton, A.: Modelling asymmetric exchange rate dependence. Int. Econ. Rev. **47**(2), 527–556 (2006)
8. Tang, K., Xiong, W.: Index investment and the financialization of commodities. Financ. Anal. J. **68**(6), 54–74 (2012)

Integrating Consumption and Reserve Strategies for Large Consumers in Electricity Markets

Nigel Cleland, Golbon Zakeri, Geoff Pritchard, and Brent Young

1 Introduction

Electricity Markets have become prevalent in a multitude of countries and jurisdictions. In most countries the structure is for privately held companies to compete with one another in order to serve an inelastic load. This approach has led to some notable failures, California in Summer 2000, but also increased efficiency of investment and operations. It is accepted that electricity markets cannot operate effective without demand side participation (DSP) [6]. Many researchers are focusing upon smart grids and retail technology which will eventually yield benefits, but more immediate progress is at hand by ensuring large consumers participate effectively. Pre-existing time of use pricing and a willingness to spend capital place them as the ideal consumers for implementing DSP. A brief experiment, using four large consumers showed that potential savings of $30 million NZD/year could result from them shaving 10 % of load during the twenty highest priced days each year.

Nodal pricing has been thoroughly studied previously in [2, 10, 15–17, 19] although the interaction of energy and reserve prices has received less attention with [2–5, 9, 10, 18] providing a good overview. The papers demonstrate the mechanisms by which the energy and reserve prices are intertwined in the Optimal Power Flow (OPF). We note that units can be dispatched non-intuitively, including out of (naive) merit order due to security considerations. We will describe the specific security

N. Cleland (✉) • G. Zakeri • G. Pritchard • B. Young
University of Auckland, Auckland, New Zealand
e-mail: nigel.cleland@auckland.ac.nz; g.zakeri@auckland.ac.nz; g.pritchard@auckland.ac.nz; b.young@auckland.ac.nz

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_4

requirements for NZ [1] and demonstrate how this coupling can occur in various ways. Any participant seeking to understand their impact upon the market needs to account for these interactions.

2 Interaction of Energy and Reserve Offers

In this paper we present a fully detailed optimal power flow dispatch software used in New Zealand to provide price distributions for a large consumer of electricity, under various operating conditions. We will utilize a representation of the New Zealand grid dispatch model called vectorised Scheduling Pricing and Dispatch (vSPD)[13] developed by the Electricity Authority. This formulation includes all energy and reserve constraints along with the full transmission network and is used to simulate final pricing. vSPD can be computationally expensive, other, more simplified models fail to capture nodal prices accurately. We outline the specifics of reserve procurement, show how they can interact with price and establish the effect on a large consumer. We then describe our optimisation model for large consumers noting that we have not considered the use of aggregators.

Reserve Procurement in New Zealand is through a 250 node co-optimised electricity network with primary (6s, FIR) and secondary (60s, SIR) reserve co-optimised with energy across two HVDC interconnected AC subnetworks. N-1 is maintained by securing sufficient reserve to cover the largest (island) risk setter, either a CCGT (400 MW) or the HVDC connection (up to 700 MW), transmission and losses are ignored in the reserve dispatch. Offers to the market follow hockey stick [11] curves and thus prices are heavily sensitive to any fluctuation in aggregate demand.

Interruptible Load (IL), Partially Loaded Spinning Reserve (PLSR) and Tail Water Depressed Spinning Reserve (TWDSR) all provide reserve in the NZEM. IL from grid connected industrial companies, PLSR and TWDSR from generation units with Hydro units being of particular importance. Dispatch from these units must satisfy the three inverse bathtub constraints [7, 8] A minimum energy reserve ratio, maximum reserve capacity and nameplate capacity constraint all limit the dispatch. In New Zealand this leads to two considerations, the first the dispatch of each unit must be feasible and aggregate reserve cleared must be larger than the risk setter.

Interactions of Energy and Reserve in OPF are the focus of [4, 5] and we remind the reader of them here. We lay out a simplified version of the OPF problem, ignoring losses (although present in vSPD) as these are understood [17] and make no essential difference to our points. Consider the (Primal) POPF and accompanying (Dual) DOPF where firms bid step function energy and reserve stacks.

$$
\begin{aligned}
&[POPF]\min \quad p_g^T g + p_r^T r && [DOPF]\max \quad d^T + R^T\omega + G^T\epsilon + F^T(\tau^+ + \tau^-) \\
&\text{st. } Mg + Af = d \; [\pi] && \text{st.} \quad M^T\pi + \epsilon - K\kappa + \lambda^1 \le p_g \quad [g] \\
&r + g \le G \quad [\epsilon] && \omega + \epsilon + \kappa + E\lambda^1 \le p_r \quad [r] \\
&r - Kg \le 0 \quad [\kappa] && A^T\pi + \tau^+ - \tau^- - B^T\lambda^2 + L^T\alpha = 0 \; [f] \\
&Er - g \ge 0 \; [\lambda^1] && \omega, \epsilon, \tau^{\pm}, \kappa \le 0 \\
&Hr - Bf \ge 0 \; [\lambda^2] && \lambda^1, \lambda^2 \ge 0 \\
&r \le R \quad [\omega] && \\
&|f| \le F \quad [\tau^{\pm}] && \\
&Lf = 0 \quad [\alpha] && \\
&r, g \ge 0 &&
\end{aligned}
$$

Here g and r and d are vectors of dispatched generation, reserve and demand while, f is the vector of flows with the objective being to minimise costs. p_g and p_r are the vectors of energy and reserve prices. M is a matrix mapping generation offers to nodes, A the node-arc incidence matrix. G is a vector of the total unit capacities, K a vector specifying the ratio between generation and reserve whilst R is the vector of reserve offer limits. E is a mapping that takes the vector of reserves and maps it into reserve zones. In the dual the π and λ are noteworthy in that they refer to the nodal energy and reserve prices accordingly. The second, third and sixth constraints of the primal specify the reverse bathtub constraints. The fourth and fifth constraints specify that reserves are procured to cover the largest risk of failure setting the reserve price. The remaining constraints are standard ensuring transmission capacity and Kirchoff's law are complied with. In the dual it is self evident that energy (π) and reserve (λ) prices are linked with ϵ, κ and ω the shadow prices of the three reverse bathtub constraints.

We Consider Case Studies as it is evident from the dual that there is interaction between energy (π) and reserve prices (λ_1 and λ_2). We illustrate this below as outlined in Fig. 1 and Table 1 using examples inspired by situations frequently encountered in the NZEM.

In Case A we consider a single node market without transmission with two generators who must meet demand. Reserve is provided by a third, reserve only, plant and thus the reserve price is, $-\lambda_1 = p_{r,1}$. A security constraint is binding upon the marginal generator, the marginal cost for energy becoming the sum of generation and security costs, $\pi_1 = p_{g,1} - \lambda_1$, co-optimisation influencing the price.

In Case B we use a two node model to illustrate security constrained transmission lines. Each node has one generator and one (independent) reserve provider. In this situation demand at node two is met by importing energy from node one across

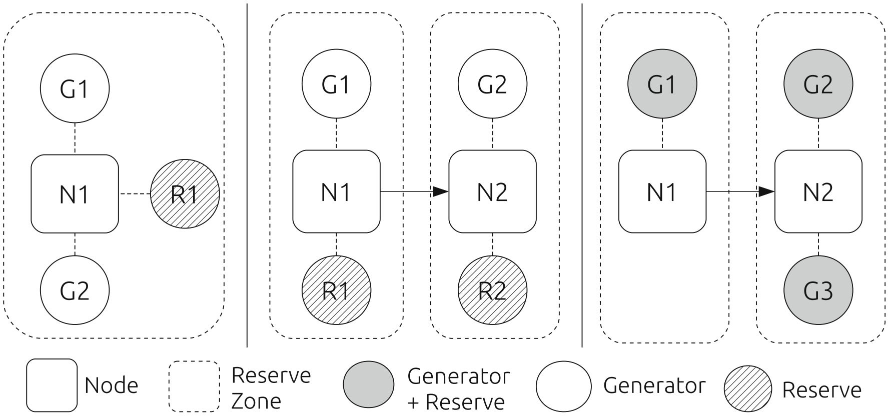


Fig. 1 Depiction of the networks used in the three theoretical cases considered. From *left* to *right* cases A, B and C. Note the separation of Generation and Reserves offers in the first scenarios, whilst they are from the same units in the third

Table 1 Case study results and information

	A	B	C		A	B	C
Demand parameters				Optimal prices			
d_1	350	50	50	π_1	30.01	0.01	0.01
d_2	–	300	310	π_2	—	45.01	670.0033
				λ_1	30	0	0
				λ_2	—	45	669.9933
				κ_2	—	—	659.9933
Offer parameters (Price, Quantity)				Optimal dispatch			
g_1	0.01, 400	0.01, 400	0.01, 300	g_1	350	350	153.3333
g_2	100, 400	100, 400	1000, 50	g_2	0	0	6.6667
g_3	–	–	10, 300	g_3	—	—	200
r_1	30, 400	30, 400	1, 300	r_1	350	0	0
r_2	–	45, 400	10, 50	r_2	—	300	3.3333
r_3	–	–	0.01, 300	r_3	—	—	100

the security constrained transmission line. We remove the requirement for sufficient reserve to clear generation for clarity. The reserve price at node two is given by the marginal unit, $-\lambda_2^2 = p_{r,2}$, with the energy price the cost of exporting an additional MW from the other node $\pi_2 = \pi_1 - \lambda_2$. This case is interesting as it explicitly links the energy prices between the two reserve zones by the reserve constraint.

In Case C we consider constrained energy and reserve dispatch at a single unit. Considering the proportionality constraint, $r - kg \leq 0$ with a k value of 0.5 we implement a constrained situation. In this case neither the energy nor reserve offers at the constrained node may be found upon the energy stack. Instead, the dual price,

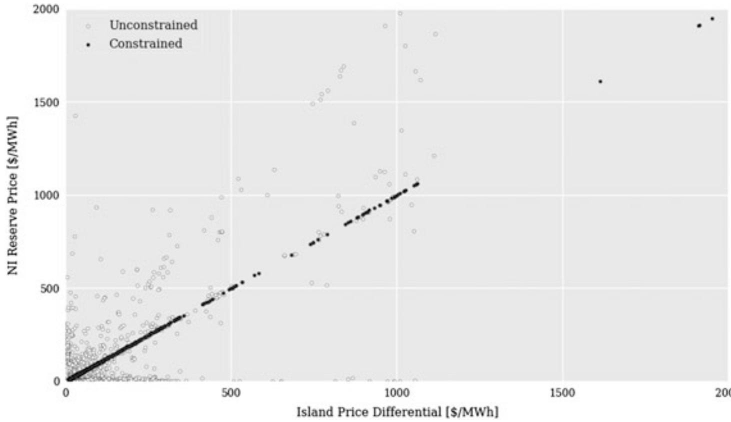


Fig. 2 Link between energy and reserve prices in the New Zealand Market, reserve constraint binding upon HVDC transfers Northward, Island Price differential refers to the difference in prices between the islands

κ is introduced into the marginal pricing equation with the final pricing at node two given by (1) with the ratio at generator two significantly affecting price.

$$\pi_2 = \frac{1}{1 + k_{g,2}} p_{g,2} + \frac{k_{g,2}}{1 + k_{g,2}} (\pi_1 + p_{r,2}) \tag{1}$$

Considering the Case of a Price Sensitive Consumer who also participates in the reserve market. This consumer receives revenue from it's IL offers, and given the price coupling behaviour as shown in Fig. 2 can often be paying a reduced effective price. This disparity between the nominal and the effective energy price forms the basis of our desire to use the simulation model. A naive optimiser, concerned with price only would reduce load in many of these situations. However the optimal decision is to continue operating as revenue from the reserve market compensates the high price.

3 Boomer Consumer

Demand side participation is identified as a key feature leading to an efficiently functioning electricity market [6, 12]. The signal for DSP is the wholesale spot price which is sensitive to demand due to the hockey stick offer stack. This can result in savings for both reduced and consumed load as price decreases. To understand DSP it is thus imperative to understand the impact of consumption upon the wholesale price.

We have demonstrated that energy and reserve price coupling can and does occur. We have developed software that utilizes vSPD [13]. For given base supply

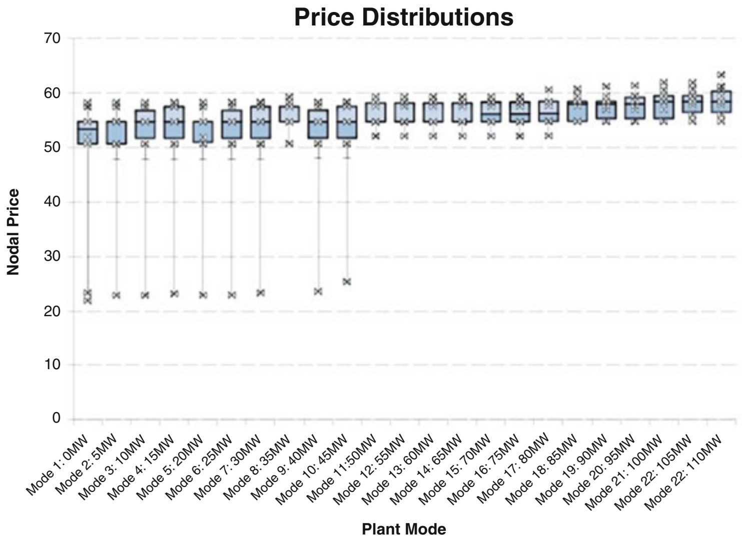


Fig. 3 Price Distributions over different modes of operation as determined using Boomer Consumer

scenarios the principle uncertainty lies in overall demand for electricity and (to a lesser extent) the distribution amongst the nodes. We use two multiplicative scale factors (NI and SI) and sample from a joint lognormal distribution of historical (fitted) data. Using vSPD a number of feasible plant consumption levels are simulated against the uncertain demand and supply scenarios to create a distribution of prices as shown in Fig. 3.

Integrating IL with DSP in reserve markets is a difficult challenge. For a consumer with a given level of energy consumption, it may be possible to offer some or all of this load as reserve. This benefits the consumer in two ways, directly earning revenue and potentially depressing energy prices by freeing up plant capacity.

The problem of offering IL in quantities great enough to (potentially) move market prices is quite similar to the one facing generators offering spot energy into the pool market. In each case, one wishes to construct a multi-tranche offer which maximizes the expected benefit of the market outcome, taking into account any natural or financial hedges one may possess. The energy-offering version of this problem was considered in [14], and we follow a similar methodology here. In essence, the (quantity, price) plane is subdivided into a finite grid of rectangular cells, and the class of admissible offer curves (supply functions) is restricted to those which follow the edges of cells in this grid. The expected value of such a curve decomposes into the sum of terms corresponding to dispatch on edges of cells (horizontal or vertical line segments); this allows the offer-optimization problem to be efficiently solved via a dynamic-programming method.

4 Conclusions

In this paper we present the development of a simulation model which assesses the impact of load consumption and reserve offers on nodal energy prices. This simulation is based upon an improved theoretical understanding of the NZEM and how reserve couples with energy prices. The model simulates a number of scenarios against uncertain demand to create an expected distribution of prices at different plant consumption levels. The site can integrate this into their decisions, along with pertinent internal factors such as the state of production, to determine their consumption level. The model explicitly integrates reserves, an improvement upon naive price based situations which would cause inappropriate curtailment. We conclude that such models are an effective tool for determining the optimal level of consumption under uncertainty in a complex market.

References

1. Alvey, T., Goodwin, D., Ma, X., Streiffert, D., Sun, D.: A security-constrained bid-clearing system for the New Zealand wholesale electricity market. IEEE Trans. Power Syst. **13**(2), 340–346 (1998). http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=667349
2. Arroyo, J., Galiana, F.: Energy and reserve pricing in security and network-constrained electricity markets. IEEE Trans. Power Syst. **20**(2), 634–643 (2005). doi:10.1109/TPWRS.2005.846221
3. Bautista, G., Quintana, V., Aguado, J.: An oligopolistic model of an integrated market for energy and spinning reserve. IEEE Trans. Power Syst. **21**(1), 132–142 (2006). doi:10.1109/TPWRS.2005.860909
4. Bautista, G., Anjos, M., Vannelli, A.: Formulation of oligopolistic competition in AC power networks: an NLP approach. IEEE Trans. Power Syst. **22**(1), 105–115 (2007). doi:10.1109/TPWRS.2006.888986
5. Bautista, G., Anjos, M., Vannelli, A.: Numerical study of affine supply function equilibrium in AC network-constrained markets. IEEE Trans. Power Syst. **22**(3), 1174–1184 (2007). doi:10.1109/TPWRS.2007.901299
6. Borenstein, S., Jaske, M., Rosenfeld, A.: Dynamic pricing, advanced metering, and demand response in electricity markets. Center for the Study of Energy Markets (2002). http://escholarship.org/uc/item/11w8d6m4.pdf
7. Chakrabarti, B.: Dispatch and pricing model with network. Wellington, NZ (2006). http://www.systemoperator.co.nz/sites/default/files/bulk-upload/documents/dispatch-and-pricing-model-20061011.pdf
8. Chakrabarti, B.: Multiple security constrained energy dispatch model. Dekalb, Illinois (2007). http://www.systemoperator.co.nz/sites/default/files/bulk-upload/documents/multiple-security-constrained-model-07.pdf
9. Chattopadhyay, D.: Multicommodity spatial cournot model for generator bidding analysis. IEEE Trans. Power Syst. **19**(1), 267–275 (2004). doi:10.1109/TPWRS.2003.821436
10. Galiana, F., Bouffard, F., Arroyo, J., Restrepo, J.: Scheduling and pricing of coupled energy and primary, secondary, and tertiary reserves. Proc. IEEE **93**(11), 1970–1983 (2005). doi:10.1109/JPROC.2005.857492

11. Hurlbut, D., Rogas, K., Oren, S.: Protecting the market from "hockey stick" pricing: How the public utility commission of texas is dealing with potential price gouging. Electr. J. **17**(3), 26–33 (2004). doi:10.1016/j.tej.2004.03.001. http://www.sciencedirect.com/science/article/pii/S1040619004000259
12. Joskow, P.L.: Lessons learned from electricity market liberalization. Energy J. **29**, 9–42 (2008). http://ezproxy.auckland.ac.nz/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=aph&AN=45388868&site=ehost-live&scope=site
13. Naidoo, R.: vSPD - vectorised scheduling pricing and dispatch model. Electricity Authority, Wellington, NZ (2015). http://www.emi.ea.govt.nz/Tools/vSPD
14. Neame, P.J., Philpott, A.B., Pritchard, G.: Offer stack optimization in electricity pool markets. Oper. Res. **51**(3), 397–408 (2003). http://or.journal.informs.org/content/51/3/397.short
15. Read, E.G.: Transmission pricing in New Zealand. Util. Policy **6**(3), 227–235 (1997). http://www.sciencedirect.com/science/article/pii/S0957178797000210
16. Ring, B.J., Read, E.G.: A dispatch based pricing model for the New Zealand electricity market. In: Einhorn, M., Siddiqi, R. (eds.) Electricity Transmission Pricing and Technology, pp. 183–206. Springer, Netherlands (1996). http://link.springer.com.ezproxy.auckland.ac.nz/chapter/10.1007/978-94-009-1804-7_8
17. Schweppe, F.C., Tabors, R.D., Caraminis, M.C., Bohn, R.E.: Spot Pricing of Electricity. Kluwer Academic Publishers, Norwell, MA (1988). http://www.osti.gov/energycitations/product.biblio.jsp?osti_id=5838573
18. Wen, F., David, A.: Optimally co-ordinated bidding strategies in energy and ancillary service markets. IET Gener. Transm. Distrib. **149**(3), 331–338 (2002). doi:10.1049/ip-gtd:20020211
19. Wu, T., Rothleder, M., Alaywan, Z., Papalexopoulos, A.: Pricing energy and ancillary services in integrated market systems by an optimal power flow. IEEE Trans. Power Syst. **19**(1), 339–347 (2004). doi:10.1109/TPWRS.2003.820701

Controlled Approximation of the Stochastic Dynamic Programming Value Function for Multi-Reservoir Systems

Luckny Zéphyr, Pascal Lang, Bernard F. Lamond, and Pascal Côté

1 Introduction

The mid-term reservoir problem usually involves release and spillage decisions in an uncertain environment. Uncertainty is mainly due to natural inflows and possibly electricity demand that may significantly vary over time. Under the framework of Stochastic Dynamic Programming (SDP), a general model can be formulated as follows:

$$V_{t-1}(s_{t-1}) = \underset{u_t}{Max} \left\{ f_t(s_{t-1}, u_t) + E_{\tilde{Q}_t} \left[V_t(s_t) \right] \right\} \tag{1}$$

$$\text{S.t. } s_t = As_{t-1} - Bu_t + \tilde{Q}_t \tag{2}$$

$$(s_{t-1}, u_t, s_t) \in C_t \tag{3}$$

where V_t is the value function; s_t is a state vector that represents the reservoir levels at the end of period t, u_t is a vector of release and spillage decisions, and $\tilde{Q}_t$ is a vector of random variables, typically natural inflows, that are assumed to be serially independent. The convex set C_t may entail bounds on reservoir levels, release and spillage as well as joint state-decision constraints. Function f_t measures a revenue or electricity production. See [4, 5], and [11] for more contextual formulations.

To circumvent the so-called *curse of dimensionality*, several schemes have been proposed over the years for approximating the value function, e.g. Stochastic Dual

L. Zéphyr (✉) • P. Lang • B.F. Lamond
Operations and Decision Systems Department, Université Laval, Pavillon Palasis-Prince, 2325, Rue de la Terrasse, Québec, QC, Canada G1V 0A6
e-mail: luckny.zephyr.1@ulaval.ca; pascal.lang@fsa.ulaval.ca; bernard.lamong@fsa.ulaval.ca

P. Côté
Rio Tinto Alcan, Énergie électrique, 1954 Davis, Jonquière, QC, Canada G7S 4R5
e-mail: Pascal.Cote@riotinto.com

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_5

Dynamic Programming [8, 10], Reinforcement Learning [1, 6], Neuro-Dynamic Programming [2, 3], Approximate Dynamic Programming [9].

In classical Dynamic Programming works, the value function is computed over a regular grid. Our approach is based on a simplicial partitioning of the state space, inducing a finite grid of points. The actual value function is computed over such grid points and, under convexity assumptions, extended as lower and upper bounds over the state space's continuum. These bounds will suggest locations where the partition can be refined.

2 Initial Partition of the State Space

We assume the state space to be a hyperrectangle $A = \{y \in R^n \mid a \leq y \leq b\}$, where $a < b$ are bounded vectors, and n is the number of reservoirs. By the change of variable $x_i = \frac{y_i - a_i}{b_i - a_i}$, this hyperrectangle is mapped to the unit hypercube $P = [0, 1]^n = \{x \in R^n \mid 0 \leq x \leq e\}$, where $e = (1, \ldots, 1)^T$.

We wish to divide this hypercube into n-dimensional initial simplices. The minimal number of such simplices is $n!$ A simple algorithm known as *Kuhn triangulation* achieves this minimum [7]. Except for the choice of an opposite pair of vertices, this algorithm does not allow for any degree of freedom.

We present a more flexible approach based on the choice of an overall "center" $c \in \text{Int}\, P$. The complexity of this method far exceeds that of the Kuhn triangulation. However, this approach may be useful as many electric utilities operate few reservoirs of practical importance.

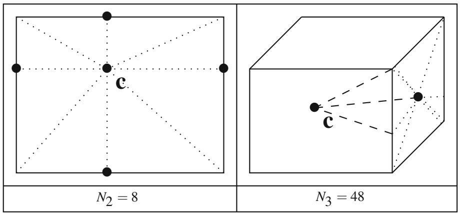
A k-dimensional face of P (hereafter k-face), with $0 \leq k \leq n$, is the hypercube that results when $n - k$ variables x_i are fixed to either bound 0 or 1. It can be shown that P contains $\binom{n}{k} 2^{n-k}$ k-faces, and overall, 3^n faces. Let F be a k-face of P. A $(k-1)$-face F' of P is called a *descendant* of F if $F' \subset F$ (i.e. one additional variable is fixed). Let $\mathscr{D}(F)$ denote the set of descendants of face F. Clearly, $\mid \mathscr{D}(F) \mid = 2k$.

We shall quasi-partition[1] each k-face F of P into a collection $\mathscr{S}_k(F)$ of N_k simplices recursively as follows. Let c_F denote the projection of center c onto face F (then $c_F \in Ri\ F$). Assume that each descendant $F' \subset F$ is already quasi-partitioned into N_{k-1} $(k-1)$-dimensional simplices $S' \in \mathscr{S}'_{k-1}(F')$. We lift each such simplex S' into k-space by constructing the set $\Sigma(S') = Conv(\{c_F\} \cup S')$[2]. Clearly, $\Sigma(S')$ is a k-dimensional simplex. Furthermore, $Ri\ \Sigma(S') \cap Ri\ \Sigma(S'') = \emptyset$ whenever $Ri\ S' \cap Ri\ S'' = \emptyset$. Finally, the union of all such lifted simplices is a cover for F. Thus, the collection $\{\Sigma(S') \mid S' \in \mathscr{S}'_{k-1}(F'), F' \in \mathscr{D}(F)\}$ is a quasi-partition of F.

[1] A quasi-partition of a set X is a finite collection of convex sets $\{C_1, \ldots, C_m\}$ such that $Ri\ C_i \cap Ri\ C_j = \emptyset\ \forall\ i \neq j$, (Ri denoting relative interior) and $\cup\, C_i = X$.

[2] *Conv* denoting convex hull.

Fig. 1 Examples of lifting



From this inductive construction, it follows that $N_k = 2kN_{k-1}$. Since P's vertices are its zero-dimensional faces while also zero-dimensional simplices, we can take them as starting family, and conclude that $N_0 = 1$. Therefore, we have $N_k = 2^k k!$. Figure 1 provides two and three dimensional illustrations.

A typical algorithm for the decomposition of the hypercube P can be outlined as follows:

$k \leftarrow 0$
Initialization : P's vertices
While $k < n$ do
  $k \leftarrow k + 1$
  For each k-face F
    For each descendant $F' \in \mathscr{D}(F)$
      For each simplex $S' \in \mathscr{S}'_{k-1}(F')$
        Construct the set $\Sigma(S')$
      End_For
    End_For
  End_For

End_While

This process is repeated until the hypercube P is decomposed into $N_n = 2^n n!$ n-dimensional simplices.

3 Iterative Division of Simplices

The vertices of the initial simplices constitute an irregular grid on which function V_{t-1} (hereafter V for short) is evaluated. Refinement of this grid may be done by iteratively dividing the existing simplices. A particular simplex S with vertices $x^1, x^2, \cdots, x^{n+1}$ is the convex hull of its extreme points:
$S = \left\{ x \middle| \begin{pmatrix} x \\ 1 \end{pmatrix} = \begin{pmatrix} X \\ e^T \end{pmatrix} \lambda, \lambda \geq 0 \right\}$, where $X = \left[x^1, x^2, \cdots, x^{n+1}\right]$ is a $n \times (n+1)$ matrix. Since these vertices are affinely independent, $\begin{bmatrix} X \\ e^T \end{bmatrix}$ is a full-rank square matrix.

Over this simplex, we have evaluations of V at its vertices: $z_i = V(x^i), i = 1, \cdots, n+1$, and a subgradient of V at each vertex: $g^i \in \partial V(x^i), i = 1, \cdots, n+1$. To evaluate the approximation gap of this function, we construct a lower and an upper bound, respectively $\underline{V}$ and $\overline{V}$. At any point $x \in S$, the difference $\overline{V}(x) - \underline{V}(x)$ provides a measure of the approximation error. If we evaluate function V at a point $y \in S$ where the estimation error is maximal, the maximal imprecision $\Delta = \overline{V}(y) - \underline{V}(y)$ may be considered an approximation gap over simplex S. Using y as division point involves subdivision of S into at least 2 subsimplices, y then contributes one additional grid point.

Under concavity assumptions, a lower bound $\underline{V}$ on V is obtained by linear interpolation from the simplex vertices: $\underline{V}(x) = z^T\lambda = z^T \begin{pmatrix} X \\ e^T \end{pmatrix}^{-1} \begin{pmatrix} x \\ 1 \end{pmatrix}, x \in S$, where $z^T = (z_1, \ldots, z_{n+1})$. The approximation gap Δ and the division point y^* may be computed by way of the following linear program:

$$\Delta = \quad \max_{\mu,y,\lambda} \quad \mu - z^T\lambda \tag{4}$$

$$S.t. \quad \mu \leq z^i + g^i(y - x^i), i = 1, \cdots, n+1 \tag{5}$$

$$y = X\lambda \tag{6}$$

$$e^T\lambda = 1 \tag{7}$$

$$\lambda \geq 0 \tag{8}$$

Simplex S is then divided into $k+1$ subsimplices, with y^* a common vertex, k being the dimension of the face on which y^* is located. To prevent discontinuity in the approximation of V, any other simplex sharing this face will be similarly devised. Figure 2 provides some two-dimensional illustrations.

The following outlines an algorithm for the division of simplices.

Repeat until $\Delta < \eta$

(i) *Choose a not divided simplex with maximal approximation gap*
(ii) *Compute $V(y^*)$ and $g(y^*)$*
(iii) *For each subsimplex, compute its Δ, y^*, $V(y^*)$ and $g(y^*)$*

η is a prescribed threshold.

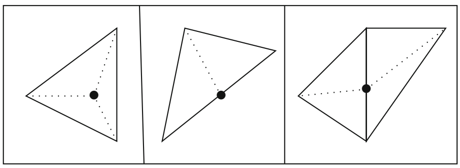


Fig. 2 Examples of subsimplices

4 Numerical Experiments

A few problem instances were generated by Monte Carlo simulations. The number of reservoirs varied from 2 to 4 (see Fig. 3).

Bounds on reservoir levels and turbine capacity as well as inflows were randomly generated from uniform distributions with supports of the form $[a, b]$. The numerical values of these parameters are summarized in Table 1.

For each run, one DP recursion was performed. We tested our method on several reservoir configurations. For each configuration, Table 2 reports the characteristics of the problems solved for each DP recursion.

Furthermore, at each iteration of the algorithm for division of simplices, the computation of the division point y^* and the imprecision gap Δ involves solution of a linear program with $2n + 2$ variables and $3n + 3$ constraints. For each run, moreover the hypercube algorithm, we performed 1000 iterations of the algorithm for division of simplices; overall, the CPU time varied approximately between 2 and 30 min.

Figure 4 reports the resulting two dimensional grid for one of the random instances, after executing the division algorithm over 65 iterations. This example gives an inkling of the imprecision that would result if a regular 8×8 grid was used instead.

Figure 5 depicts the evolution of the imprecision over one thousand iterations for 4 of those instances. In all four cases, the rate of improvement in the relative "error", while initially significant, seems to taper off over iterations. We may therefore conjecture that the algorithm's convergence rate is sublinear.

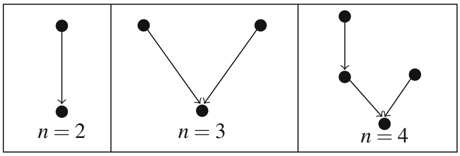


Fig. 3 Reservoir configurations

Table 1 Numerical bounds on the parameters

Parameter	a	b
$\underline{s}_i$	500	2000
$\bar{s}_i$	5000	11,000
$\underline{u}_i$	0	0
$\bar{u}_i$	2000	15,000
q_i	0	2000

Table 2 Problem characteristics

n	# of variables	# of constraints
2	78	90
3	81	98
4	84	106

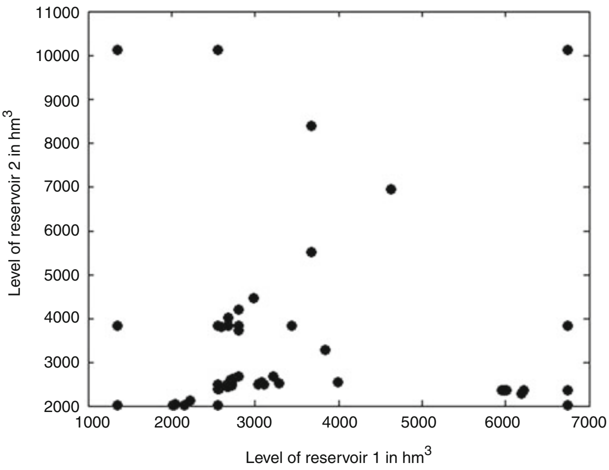


Fig. 4 Example of a grid for a model with 2 reservoirs (after 65 iterations)

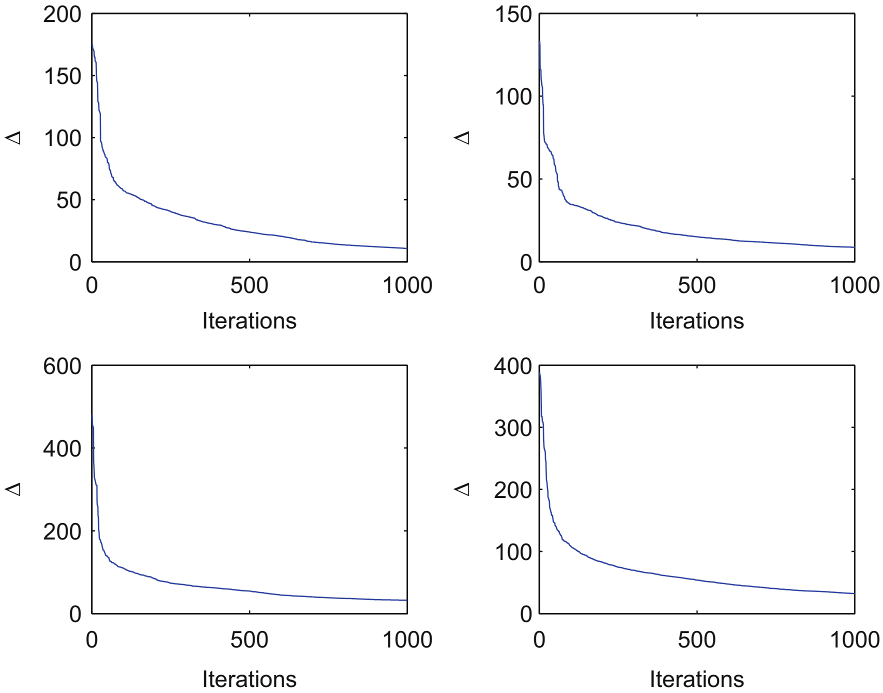


Fig. 5 Approximation gap over iterations for a model with 4 reservoirs

5 Conclusions

We have presented an approach to approximate the value function of Stochastic Dynamic Programming. Our method is based on partitioning the state space into simplices followed by iterative division of existing simplices. Refinement of the approximation is guided by lower and upper bounds on the true value function. This method can be particularly applied in the context of reservoir systems management. However, the complexity of hypercube decomposition limits its scope to less than 10 reservoirs.

References

1. Castelletti, A., Galelli, S., Restelli, M., Soncini-Sessa, R.: Tree-based reinforcement learning for optimal water reservoir operation. Water Resour. Res. **46**(9) W09507, doi:10.1029/2009WR008898 (2010)
2. Cervellera, C., Wen, A., Chen, V.C.: Neural network and regression spline value function approximations for stochastic dynamic programming. Comput. Oper. Res. **34**(1), 70–90 (2007)
3. Chandramouli, V., Raman, H.: Multireservoir modeling with dynamic programming and neural networks. J. Water Resour. Plan. Manag. **127**(2), 89–98 (2001)
4. Gil, E., Bustos, J., Rudnick, H.: Short-term hydrothermal generation scheduling model using a genetic algorithm. IEEE Trans. Power Syst. **18**(4), 1256–1264 (2003)
5. Kim, Y.O., Eum, H.I., Lee, E.G., Ko, I.H.: Optimizing operational policies of a korean multireservoir system using sampling stochastic dynamic programming with ensemble streamflow prediction. J. Water Resour. Plan. Manag. **133**(1), 4–14 (2007)
6. Lee, J.H., Labadie, J.W.: Stochastic optimization of multireservoir systems via reinforcement learning. Water Resour. Res. **43**(11), W11408 (2007)
7. Munos, R., Moore, A.: Variable resolution discretization in optimal control. Mach. Learn. **49**(2–3), 291–323 (2002)
8. Philpott, A.B., Guan, Z.: On the convergence of stochastic dual dynamic programming and related methods. Oper. Res. Lett. **36**(4), 450–455 (2008)
9. Powell, W.B.: What you should know about approximate dynamic programming. Nav. Res. Logist. **56**(3), 239–249 (2009)
10. Shapiro, A.: Analysis of stochastic dual dynamic programming method. Eur. J. Oper. Res. **209**(1), 63–72 (2011)
11. Zéphyr, L., Lang, P., Lamond, B.F.: Adaptive monitoring of the progressive hedging penalty for reservoir systems management. Energy Syst. 1–16. Published online: 20 December 2013. doi:10.1007/s12,667-013-0110-4 (2013)

A Computational Method for Predicting the Entropy of Energy Market Time Series

Francesco Benedetto, Gaetano Giunta, and Loretta Mastroeni

1 Introduction

The understanding of the dynamic behaviour of energy market time series (EMTS) is of great and crucial interest for the analysis of energy commodities. In particular, the observation of historical data as well as the analysis of their volatility (and price fluctuations) can be useful indicators of the dynamic characteristics of the series, in order to effectively perform forecasting procedures. We can address, as usual, to standard deviation (SD) as a measure of deviation from the mean, while we will use entropy as the metric for evaluating the irregularities and, hence, the predictability of a series.

There are plenty of works exploiting the concept of entropy applied to analysis of financial and energy markets time series. For example, the validity of the entropy approach for analyzing financial time series is demonstrated in [2]. Then, in [7], an empirical method for evaluating the entropy of a series is proposed, namely the approximate entropy. Recently, studies focusing on the energy market have been carried out under the entropy-based approach. An entropy analysis of crude oil price dynamics is revealed in [4], while evidences from informational entropy analysis in evaluating the efficiency of crude oil markets were discussed in [6].

In this paper, we move further by proposing an algorithm to predict the entropy regarding the future behaviour of EMTS, based on the observation of historical data. Our algorithm exploits the concept of entropy under an information theory

F. Benedetto (✉) • L. Mastroeni
Department of Economics, "Roma Tre" University, via Silvio D'Amico 77, 00145 Rome, Italy
e-mail: francesco.benedetto@uniroma3.it; loretta.mastroeni@uniroma3.it

G. Giunta
Department of Engineering, "Roma Tre" University, via Silvio D'Amico 77, 00145 Rome, Italy
e-mail: giunta@ieee.org

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_6

viewpoint, recalling the *maximum entropy theory* to evaluate the entropy estimation. In addition, our prediction is performed according to optimum prediction methods, i.e. following the least squares minimization scheme (as happens in conventional computational and engineering prediction approaches).

The remainder of this work is organised as follows. Section 2 discusses the basic frameworks about energy market-based entropy analysis. The first half of the section is dedicated to the maximum entropy theory while in the second half the approximate entropy method is briefly illustrated. Then, our proposed entropy estimator is shown in details in Sect. 2.2, with all the mathematical derivations. Section 3 contains some preliminary results and discussions about the application of our method to EMTS. Finally, our conclusions are depicted in Sect. 4.

2 Energy Market Entropy Analysis

2.1 Maximum Entropy Theory

Previous characterizations of the maximum entropy spectral density assume that the process is stationary and Gaussian. Let us now define with Cov(k) the autocovariance function of the input random series x(n) of N data, and defined as:

$$\mathrm{Cov}\,(\mathrm{k}) = \frac{1}{\mathrm{N}} \sum_{\mathrm{i}=1}^{\mathrm{N}} \mathrm{x}(\mathrm{i}) \cdot \mathrm{x}^{*}(\mathrm{i}-\mathrm{k}) - |\mu|^{2} \tag{1}$$

where $k = -N, \ldots +N$, x*(n) stands for complex conjugate, $i = 0, \pm 1, \pm 2, \ldots$, and the mean μ is expressed by

$$\mu = \frac{1}{\mathrm{N}} \sum_{\mathrm{n}=1}^{\mathrm{N}} \mathrm{x}(\mathrm{n}) \tag{2}$$

The autocovariance function can be analyzed in the transformed (frequency) domain, obtaining the power spectral density (PSD) S(ω) given by [5]:

$$\mathrm{S}(\omega) = \sum_{\mathrm{k}=-\infty}^{\infty} \mathrm{Cov}(\mathrm{k}) \cdot \mathrm{e}^{-\mathrm{j}\omega \mathrm{k}} \tag{3}$$

Now, according to [1], the entropy rate h is given by:

$$\mathrm{h} = \frac{1}{2} \ln\,(2\pi \mathrm{e}) + \frac{1}{4\pi} \cdot \int_{-\pi}^{\pi} \ln(\mathrm{S}(\omega))\mathrm{d}\omega \tag{4}$$

where ln(·) is the natural logarithm. It is now interesting to underline that the entropy of a finite segment of a stochastic process is upper-bounded by the entropy of a segment of a Gaussian random process, according to (4). This means that a white (i.e. uncorrelated) time series is characterized by the maximum entropy, i.e. it is obviously unpredictable [1]. Lower entropy values result in more predictable time series.

2.2 Proposed Maximum Entropy Estimator

Given a time series x(n) of length N samples, i.e. n = 1, 2,…N, the proposed maximum entropy estimator (MEE) works accordingly to the following steps. First, the N samples of x(n) are divided in K blocks, each of length $M = N / K$ samples. Then, the mean is estimated for each i-th block, according to the following:

$$\mathrm{Mean_i} = \frac{1}{\mathrm{M}} \sum_{\mathrm{j}=1}^{\mathrm{M}} \mathrm{x_i(j)} \tag{5}$$

where x_i (j) stands for the j-th sample of the i-th block, with $i = 1, \ldots, K$ and j = 1, …, M. Then, the mean of the i-th block is subtracted from the same i-th block, so that the blocks are zero-mean series:

$$\mathrm{y_i\,(j) = x_i\,(j) - Mean_i} \tag{6}$$

This step is required because the object of the maximum entropy analysis needs to be a zero-mean series. Now, the autocorrelation function, $C_{y,i}$ (·), of each block (i.e. of each sequence $y_i(\cdot)$, with $i = 1, 2, \ldots, K$) is evaluated according to the following:

$$\mathrm{C_{y,i}\,(k)} = \frac{1}{\mathrm{M}} \sum_{\mathrm{j}=1}^{\mathrm{M}} \mathrm{y_i(j) \cdot y_i^*(j-k)} \tag{7}$$

The autocorrelation function is symmetric and has a maximum in zero, by definition (see [5]), hence it is not more a zero mean series. But, the input of the maximum entropy analysis must be a zero-mean series. Hence, we have now to evaluate the mean of each i-th autocorrelation block, m_i, and subtract these means blockwise. These means are evaluated using Eq. (5), with $C_{y,i}(\cdot)$ instead of $x_i(\cdot)$. In practice, we are evaluating the autocovariance function of each sequence, instead of the autocorrelation, according to the following:

$$\mathrm{Cov}_{y,i}(k) = \frac{1}{M}\sum_{j=1}^{M} y_i(j) \cdot y_i^*(j-k) - |\mu_i|^2 \tag{8}$$

where y*() means complex conjugate. Now, the K autocovariance functions become the input of the optimum linear predictor of parametric order p. The outputs of the optimum linear prediction step are p prediction coefficients that are now used to estimate the autocovariance of the next block that is the block of which we want to evaluate the entropy. Let us now define with $\widehat{\mathrm{Cov}}_{y,K+1}(k)$ the predicted autocovariance sequence of the (K+1)-th block. This sequence is evaluated according to the following:

$$\widehat{\mathrm{Cov}}_{y,K+1}(k) = \sum_{b=0}^{p-1} a_b \cdot \mathrm{Cov}_{y,K-b}(k) \tag{9}$$

where $\mathrm{Cov}_{y,K-b}(k)$ are the linearly combined (K - b) previous observed blocks and a_b are the AR coefficients, evaluated according to the optimum least squares minimization algorithm, [5]. In particular, we use the Levinson-Durbin recursion to solve the least-squares formulation, often referred to as the autocorrelation method [5]. Now, the predicted autocovariance sequence is first transformed in the frequency domain, see Eq. (3), obtaining the PSD of the analyzed block. Then, the entropy of the (K+1)-th block is estimated according to the maximum entropy approach, see Eq. (4).

3 Application to Energy Market Time Series

We analyze daily prices (or observations) of two different commodities (Brent Crude Oil and WTI oil prices) in the period between May 20, 1991 and August 14, 2012. In particular, the time series were obtained from http://www.quandl.com. Figure 1 shows the annual entropy variations obtained with both the conventional approximate entropy (ApEn) technique (with m = 1, and a value of r=20 % of the SD of the series) and the new computational method for the Brent Crude Oil prices (with K = 26 and p = 10). The ApEn method estimates the entropy ex-post (i.e. we need all the samples of that current year to estimate the entropy of that year), while we evaluate the entropy ex-ante (i.e. we only need the past samples to predict the entropy of the next time interval). In full accordance with [4], it is clearly visible from Fig. 1 that some peaks of the entropy pattern coincide with the outbreak of some major events.

For example, we have clear entropy peaks in correspondence to critical events, such as the 1991 Gulf War, 2001-9-11 terrorist attacks and the Lehman Brothers bankruptcy. This suggests that major financial and socio-political events strongly affect the diversity of the Brent crude oil market. Then, Fig. 2 depicts the annual entropy variations obtained with both the two methods for the WTI Oil prices. In

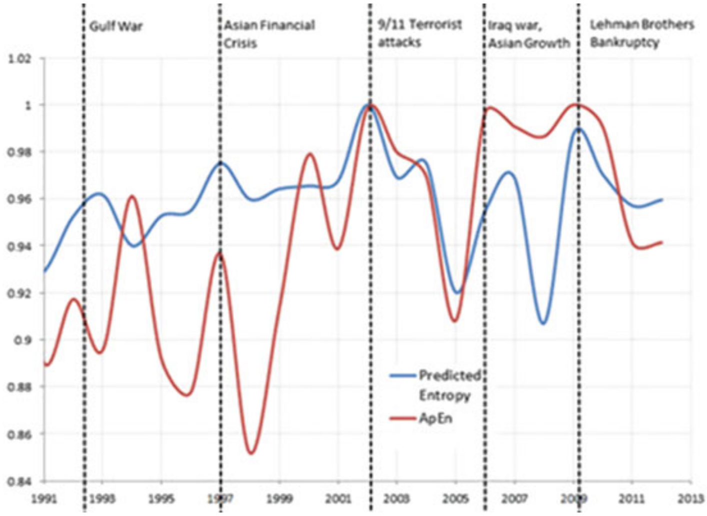


Fig. 1 Annual normalized (to 1) entropy variations of the Brent Crude oil prices

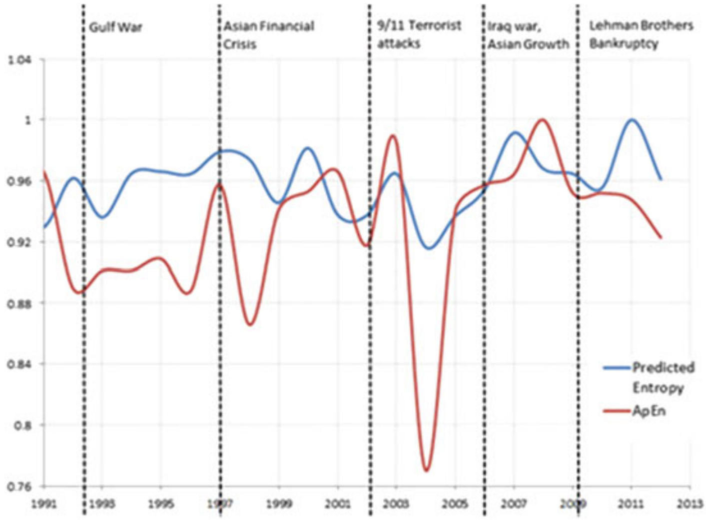


Fig. 2 Annual normalized (to 1) entropy variations of the WTI oil prices

this case, the correspondences between the entropy peaks and some socio-political events are less evident for both the methods. This is because, as also stated in [3], the WTI oil series is characterized by a more unpredictable behavior than the Brent crude oil series.

4 Conclusions

This work presents a new computational method for predicting the entropy of a time series. We have applied our prediction technique to energy market time series, matching the conventional approximate entropy estimation method. Preliminary results encourage the application of our method to energy-market analysis. Further researches will be devoted to fully characterize the performance of this new prediction method, and also its application to financial market time series.

References

1. Chol, B.S., Cover, T.M.: An information-theoretic proof of burg's maximum entropy spectrum. Proc. IEEE **72**, 1094–1095 (1984)
2. Darbellay, G.A., Wuertz, D.: The entropy as a tool for analysing statistical dependences in financial time series. Physica A **287**, 429–439 (2000)
3. Kristoufek, L., Vosvrda, M.: Commodity futures and market efficiency. Energy Econ. **42**, 50–57 (2014)
4. Martina, E., Rodriguez, E., Escarela-Perez, R., Alvarez-Ramirez, J.: Multiscale entropy analysis of crude oil price dynamics. Energy Econ. **33**, 936–947 (2011)
5. Oppenheim, A.V., Schafer, R.W.: Digital Signal Processing. Prentice-Hall, Upper Saddle River, NJ (1975)
6. Ortiz-Cruz, A., Rodriguez, E., Ibarra-Valdez, C., Alvarez-Ramirez, J.: Efficiency of crude oil markets: evidences from informational entropy analysis. Energy Econ. **41**, 365–373 (2012)
7. Pincus, S.M., Kalman, E.K.: Irregularity, volatility, risk, and financial market time series. Proc. Natl. Acad. Sci. **101**, 13709–13714 (2004)

Part II
Logistics

Demand Uncertainty for the Location-Routing Problem with Two-dimensional Loading Constraints

Thiago Alves de Queiroz, José Fernando Oliveira, Maria Antónia Carravilla, and Flávio Keidi Miyazawa

1 Introduction

King and Mast [7] pointed out that the final cost of the goods can increase between 10 and 15 % depending on the supply chain infrastructure. In order to reduce such cost, integrated decisions from strategic, tactical and operational levels must be considered when planning and designing logistic systems. A problem that attains these three levels is the Location-Routing Problem (LRP), in which decisions from the strategic (where to locate depots), tactical (which customers to serve from each depot) and operational levels (decide the routing plan) are taken simultaneously.

Belenguer et al. [2] presented a branch-and-cut algorithm to solve the LRP, which is strengthened by valid inequalities and separation algorithms. A branch-and-cut-and-price approach was developed in [1] allowing to solve instances with up 199 customers.

In this paper, we deal with the LRP with two-dimensional loading constraints (2L-LRP) and demand uncertainty, an integrated problem without any reference in the literature, through an integer programming model. In this case, the customers' demand are pallets that must be arranged inside the vehicles. Demand uncertainty is described by a scenario approach and appears due to the volatility in the markets [3].

In Sect. 2, we formally describe the problem and present the integer model. In Sect. 3, a computational experiment over one instance adapted from a real case study is detailed. Finally, conclusions and directions for future work are given in Sect. 4.

T.A. de Queiroz (✉) • J.F. Oliveira • M.A. Carravilla
INESC TEC, Faculty of Engineering, University of Porto, 4200465 Porto, Portugal
e-mail: taq@ufg.br; jfo@fe.up.pt; mac@fe.up.pt

F.K. Miyazawa
Institute of Computing, State University of Campinas, 13083852 Campinas, SP, Brazil
e-mail: fkm@ic.unicamp.br

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_7

2 Problem Definition and Integer Formulation

In the 2L-LRP we have: a set of possible depot locations I, in which each $i \in I$ has weight storage capacity b_i and opening cost O_i; a set of customers J, where each $j \in J$ has a set R_j of rectangular items, in which the total area of items in R_j is aR_j and the total weight is dR_j. And, each item $r \in R_j$ has length l_{jr}, width w_{jr}, area a_{jr} and weight d_{jr}; a set of identical vehicles, each one with weight capacity Q, rectangular surface area with dimensions (L, W) and fixed cost F when used; and, an undirected graph $G = (V, E)$, with $V = I \cup J$ representing the set of vertices and E the set of edges, each edge e with a traveling cost c_e. The graph is complete for the connections between customer-customer and depot-customer, however there is no edge for the relation depot-depot.

A solution of the 2L-LRP consists in opening a subset of depots, from which routes are established respecting the vehicle capacity and serving the customers. The number of routes, for each depot, is limited by the respective depot capacity, and each customer is visited exactly once. Each route starts and finishes at the same depot, and is formed by a sequence of visited customers, such that their items can be arranged without overlapping inside the vehicle's rectangular surface.

The demand uncertainty is tackled by a scenario approach in which each scenario s of a set of scenarios $\mathscr{P}$ represents different demands for the customers and has probability p_s of occurrence, such that $\sum_{s \in \mathscr{P}} p_s = 100\,\%$. In this way, we can construct solutions that are robust in face of the market's volatility, and simultaneously effective when planning the supply chain.

The integer model for the 2L-LRP is described in the integer formulation below. The notation used is the following: $\delta(S)$ represents the edges with one end-node in S and the other in $V - S$; $D_s(S) = \lceil \frac{\sum_{j \in S_s} \sum_{r \in R_j} d_{jr}}{Q} \rceil$ is a lower bound on the number of vehicles necessary to supply the weight in $S \subseteq J$, in accordance with scenario $s \in \mathscr{P}$; $A_s(S) = \lceil \frac{\sum_{j \in S_s} \sum_{r \in R_j} a_{jr}}{A} \rceil$, in which A denotes the vehicles' rectangular surface area.

The decision variables are: $y_i = 1$, indicating that a depot is open at location $i \in I$; $x_{ijs} = 1$ when a depot at $i \in I$ serves customer $j \in J$ in scenario $s \in \mathscr{P}$; and, $w_{jks} = 1$ imposing that edge $\{j, k\} \in E$, in scenario $s \in \mathscr{P}$, is traversed exactly once. Routes that serve only one client, called *return trips* in [2], are modeled by considering the duplicated set $I' = I$, so $V = V \cup I'$, and new edges $\{i', j\}$ for $i' \in I'$ and $j \in J$ are added in E. Note that the decision to open a depot must be performed observing all scenarios in $\mathscr{P}$, since it represents a long term decision (strategic one) whose cost is significantly greater than the other ones.

The objective function of the integer formulation aims to minimize the overall cost given by the fixed cost of opening depots plus the cost associated with the probability of occurrence of each scenario. And, for each scenario, there is the fixed cost of vehicle usage, related with the number of routes, and the total cost of the routes. Constraints (1) ensure that each customer, in each scenario, is served by exactly one depot, while constraints (2) impose that the capacity of each depot must be respected. Constraints (3) consider that customers can only be served from open depots, and (4) are the degree constraints for the customers, for each scenario. Constraints (5) impose that there is a minimum number of routes starting from each depot, in each scenario, in order to serve the customers' weight demand.

The global minimum number of routes that must be established to serve all the customers' demand is guaranteed in constraints (6). And, if a depot is opened, it has to serve at least one customer as defined in (7). It is worth to mention that $A_s(S)$ is just a continuous lower bound of the two-dimensional bin packing problem. Nevertheless, we need to solve this problem in order to get the precise number of bins/vehicles really necessary to arrange all items in S. Similarly for $D_s(S)$ in the one-dimensional case.

The capacity constraints for the vehicles are in (8), and constraints (9) ensure that there is a path connecting each depot to its customers. Moreover, if there is an edge connecting a given customer with another one, this customer can not be in a return trip as pointed in (10), while constraints (11) consider the opposite. Constraints (12) impose that a customer k must be served by the same depot i which serves customer j if k is connected with j. Constraints (13) and (14) make the correspondence between variables x_{ijs} and w_{ijs} relating the customer-depot. To handle the two-dimensional packing problem, constraints (15) eliminate routes in which the respective packing is not feasible. Finally, constraints (16)–(18) impose that all the variables are binary.

The number of constraints (8), (9) and (15) may be very large, so they are added as cutting planes and detected with specific separation algorithms. The algorithms for (8) and (9), applied both on integer and fractional solutions, are based on the computation of the Gomory-Hu tree, similar to that in [6]. So for each min $s - t$ cut, for $s \in I$ and $t \in J$, we check the violation of such constraints assuming S with all nodes of the t-component. Although constraints (9) can be efficiently separated with this procedure, we also used for (8) the separation strategy proposed in [8] for the rounded capacity inequalities when dealing with the capacitated vehicle routing problem.

On the other hand, constraints (15) are checked only when an integer feasible solution is found, since testing the feasibility of a packing is more time consuming, and in fact it is an NP-hard problem [5]. For this task, we use the constraint programming based approach proposed in [4], and modify it to take into consideration the sequence in which customers are visited in the route. This means that items from a given customer are accessible when the unloading operation occurs, namely multi-drop requirements [9].

$$\min \quad \sum_{i \in I} O_i y_i + \sum_{s \in \mathscr{P}} p_s \left(\frac{F}{2} \sum_{i \in I \cup I'} \sum_{j \in J} w_{ijs} + \sum_{\{i,j\} \in E} c_{ij} w_{ijs} \right)$$

subject to :

$$\sum_{i \in I} x_{ijs} = 1, \qquad \forall\, j \in J,\ \forall\, s \in \mathscr{P} \tag{1}$$

$$\sum_{j \in J} d_j x_{ijs} \leq b_i y_i, \qquad \forall\, i \in I,\ \forall\, s \in \mathscr{P} \tag{2}$$

$$\sum_{s \in \mathscr{P}} x_{ij} \leq y_i |S|, \qquad \forall\, i \in I,\ \forall\, j \in J \tag{3}$$

$$\sum_{e \in \delta(j)} w_{es} = 2, \qquad \forall\, j \in J,\ \forall\, s \in \mathscr{P} \tag{4}$$

$$\sum_{j \in J} (w_{ijs} + w_{i'js}) \geq 2 \frac{\sum_{j \in J} d_{js} x_{ijs}}{Q}, \qquad \forall\, i \in I,\ \forall\, s \in \mathscr{P} \tag{5}$$

$$\sum_{i \in I \cup I'} \sum_{j \in J} w_{ijs} \geq 2 \max\{D_s(J); A_s(J)\}, \qquad \forall\, s \in \mathscr{P} \tag{6}$$

$$\sum_{s \in \mathscr{P}} \sum_{j \in J} x_{ijs} \geq y_i, \qquad \forall\, i \in I \tag{7}$$

$$\sum_{e \in \delta(S)} w_{es} \geq 2 \max\{D_s(S),\ A_s(S)\}, \qquad \forall\, S \subseteq J,\ \forall\, s \in \mathscr{P} \tag{8}$$

$$\sum_{e \in \delta(S)} w_{es} \geq 2(x_{ijs} + y_i - 1), \qquad \forall\, S \subseteq J,\ \forall\, j \in S,\ \forall\, i \in I,\ \forall\, s \in \mathscr{P} \tag{9}$$

$$\sum_{i' \in I'} w_{i'js} \leq 2 - \Big(\sum_{k \in J} w_{jks} + \sum_{i \in I} w_{ijs}\Big), \qquad \forall\, j \in J,\ \forall\, s \in \mathscr{P} \tag{10}$$

$$\sum_{k \in J} w_{jks} \geq 2 - (w_{ijs} + w_{i'js}), \qquad \forall\, j \in J,\ \forall\, i \in I:\ i' = i\ \in I',\ \forall\, s \in \mathscr{P} \tag{11}$$

$$w_{jks} + x_{ijs} \leq 1 + x_{iks}, \qquad \forall\, j,k \in J,\ \forall\, i \in I,\ \forall\, s \in \mathscr{P} \tag{12}$$

$$w_{i'js} \leq w_{ijs}, \qquad \forall\, j \in J,\ \forall\, i \in I:\ i' = i\ \in I',\ \forall\, s \in \mathscr{P} \tag{13}$$

$$w_{i'js} + w_{ijs} \leq 2x_{ijs}, \qquad \forall\, j \in J,\ \forall\, i \in I:\ i' = i\ \in I',\ \forall\, s \in \mathscr{P} \tag{14}$$

$$\sum_{e \in R} w_{es} \leq |R| - 1, \qquad \forall\, R \in \mathscr{R}_s,\ \forall\, s \in \mathscr{P} \tag{15}$$

$$y_i \in \{0, 1\}, \qquad \forall\, i \in I \tag{16}$$

$$x_{ijs} \in \{0, 1\}, \qquad \forall\, i \in I,\ \forall\, j \in J,\ \forall\, s \in \mathscr{P} \tag{17}$$

$$w_{es} \in \{0, 1\}, \qquad \forall\, e \in E,\ \forall\, s \in \mathscr{P} \tag{18}$$

3 Computational Study

In order to verify the quality of the model, we used it to solve one instance adapted from [3], a real case based on an European supply chain. The plants and warehouses are possible depot locations, so $I = 8$, while there are $J = 30$ customers (retailers and markets). For each depot i, its capacity b_i is given by the number of technologies (according to [3] each plant has 12, and each warehouse has 6) multiplied by the factor α^2. If the depot is a plant its cost is the sum of the fixed cost of each technology. Otherwise, the cost is the fixed cost of the warehouse, plus its variable cost multiplied by its capacity, added to the sum of the fixed cost of each technology.

Each vehicle transports one forty-foot container, with rectangular surface equal to $A = L \times W = 2358 \times 12032\,\text{mm}^2$ and max payload of $Q = 26.600\,\text{kg}$. Moreover, the values of L and W are divided by the factor α. The fixed cost F of using a vehicle is the inventory cost (0.3€) multiplied by the rectangular surface area. The cost c of each edge corresponds to the fixed cost of transportation (300€) plus the variable cost of 0.1€ multiplied by the distance between the vertices, given in km.

In order to create the demand of each customer we consider the dimensions of standard pallets divided by the factor α. The weight/payload d of each pallet is equal to its area multiplied by the correctness factor β. As [3] did not consider two-dimensional items, we randomly determined the number of items R_j and assigned them to the pallets of each customer j. The size of R_j varies between 5 and 10. We assumed $\alpha = 100$ and $\beta = 10$, and considered only the integer part of the resulting values.

Following [3], three scenarios are considered: (i) realistic, with $p_1 = 50\,\%$, so there is no change in the customers' demand; (ii) optimistic, with $p_2 = 25\,\%$, in which the demand, that is, the total number of items increases around 15 %; and (iii) pessimistic, with $p_3 = 25\,\%$, which considers a decrease by almost 15 % in customers' demand. Figure 1 illustrates the result returned after the solver reached the time limit of 24 h considering a computer with 1.90 GHz Intel Xeon E5-2420 CPU, 32 GB of RAM memory, Gurobi Optimizer 5.6.2 (for the integer formulation) and IBM ILOG CP Optimizer 12.5 (for the constraint programming algorithm). The time limit of 2 s was used in each call to the constraint programming algorithm, but such algorithm always returned a solution before reaching this time limit.

This solution, with a gap of 13.2 %, has value of the objective function equal to 38,555.75. The number of user cuts inserted over the branch-and-bound tree is of 115,865. The edges marked as added and deleted in scenarios #2 and #3 show the change in the routes when the customers' demand increases and decreases, respectively, in comparison to scenario #1, the realistic one.

Comparing the solution for each scenario in Fig. 1, the realistic one, scenario #1, requires 9 routes, while in #2 it is increased to 11, and decreased to 9 in scenario #3. Note that the solution is in accordance with the characteristics of each scenario. Although the CPU time can be considered high at a first sight, the problem under consideration has strategic and tactical decisions. Moreover, to the best of our knowledge, there is no exact algorithm neither integer formulations available for the 2L-LRP in the literature, including the version with demand uncertainty.

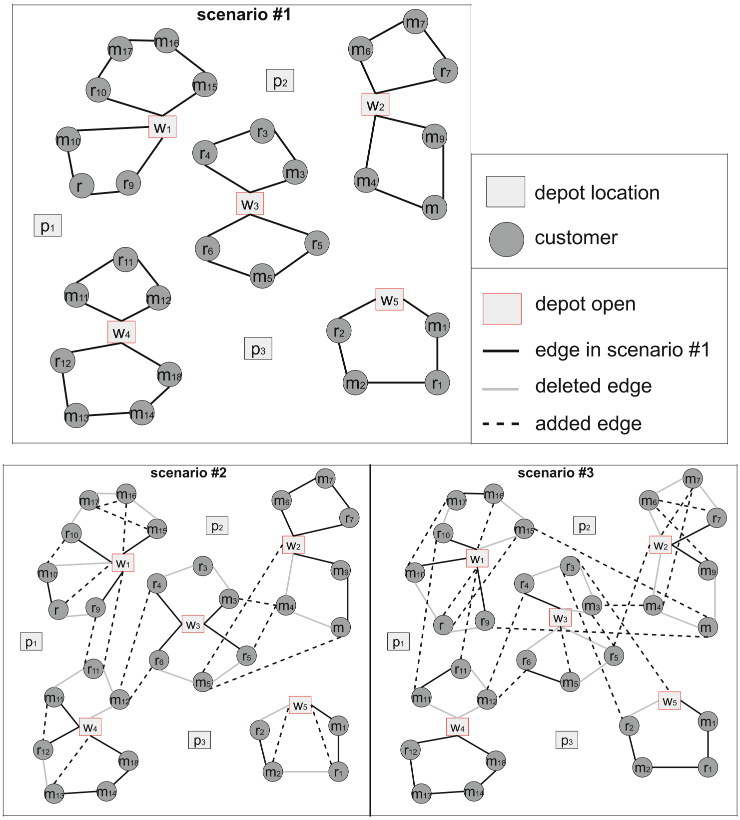


Fig. 1 Final solution in which *p* stands for plant, *w* for warehouse, *m* for market and *r* for retailer. The depots/customers' position in the figure does not correspond and are not related with those given in the instance

4 Concluding Remarks

We proposed an integer formulation for a new variant of the location-routing problem. The computational study over one instance adapted from a real-world problem shows that the integer formulation is suitable for small instances, since while operational decisions, as the determination of vehicle routes, have to be taken quickly, the location of depots or the link between customers and depots are tactical and even strategical decisions and therefore have a larger timespan to be taken.

After all, we observe that there is room for improvements by considering new separation algorithms and valid inequalities, as well as by introducing good lower bounds instead of checking the packing feasibility every time.

Acknowledgements This research was partially financed by CNPq, FAPESP and FAPEG.

References

1. Baldacci, R., Mingozzi, A., Calvo, R.W.: An exact method for the capacitated location-routing problem. Oper. Res. **59**(5), 1284–1296 (2011)
2. Belenguer, J.M., Benavent, E., Prins, C., Prodhon, C., Calvo, R.W.: A branch-and-cut method for the capacitated location-routing problem. Comput. Oper. Res. **38**, 931–941 (2011)
3. Cardoso, S., Barbosa-Póvoa, A.P., Relvas, S.: Design and planning of supply chains with integration of reverse logistics under demand uncertainty. Eur. J. Oper. Res. **226**, 436–451 (2013)
4. Clautiaux, F., Jouglet, A., Carlier, J., Moukrim, A.: A new constraint programming approach for the orthogonal packing problem. Comput. Oper. Res. **35**, 944–959 (2008)
5. Fekete, S.P., Schepers, J., van der Veen, J.: An exact algorithm for higher-dimensional orthogonal packing. Oper. Res. **55**(3), 569–587 (2007)
6. Iori, M., Salazar-Gonzalez, J.J., Vigo, D.: An exact approach for the vehicle routing problem with two-dimensional loading constraints. Transp. Sci. **41**, 253–264 (2007)
7. King, G.F., Mast, C.F.: Excess travel: causes extent and consequences. Transp. Sci. Rec. **1111**, 126–134 (1997)
8. Lysgaard, J., Letchford, A.N., Eglese, R.W.: A new branch-and-cut algorithm for the capacitated vehicle routing problem. Math. Program. Ser. A, B **100**(2), 423–445 (2004)
9. Queiroz, T.A., Miyazawa, F.K.: Two-dimensional strip packing problem with load balancing, load bearing and multi-drop constraints. Int. J. Prod. Econ. **145**, 511–530 (2013)

Location Game and Applications in Transportation Networks

Vladimir Mazalov, Anna Shchiptsova, and Yulia Tokareva

1 Introduction

This paper studies a non-cooperative game in a transportation graph. Consider a market where the consumers are distributed in the vertexes of the transportation graph $G(V, E)$. The edges of the graph are transportation links (railways, highways, airlines, etc.). The vertexes are the hubs (bus stops, airports, railway stations, etc.). The demand is determined by the flow of passengers.

There are n companies (players) who make a service in this market. A service is possible only if there is a link $e_j \in E$ between two vertexes in graph $G(V, E)$. The demand is determined by the number of consumers in vertexes $v_1, v_2 \in V$ connected by the link e_j

$$d(e_j) = d(v_1, v_2), \qquad e_j = (v_1, v_2).$$

Assume, that player i has m_i units of a resource. He distributes the resource among the links in graph $G(V, E)$. Suppose, that each player i distributes m_i units of the resource and forms the transportation network E^i which is a subset of the links in graph $G(V, E)$.

The demand on the link e_j is distributed between players. Each player presents the service for the part M_{ij} of the consumers on this link. Players announce the prices for the service on the link e_j. The part of customers which prefer the service

V. Mazalov (✉) • A. Shchiptsova
Institute of Applied Mathematical Research, Petrozavodsk, Russia

Y. Tokareva
Transbaikal State University, Chita, Russia

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_8

of player i depends of the price p_{ij} and the prices of other players on this link

$$M_{ij} = M_{ij}(p_{ij}, \{p_{rj}\}_{r\in N_j\setminus\{i\}}), \quad |M_{ij}| \le 1,$$

where N_j—number of the rival players on the link e_j.

The number of consumers who prefer the service i on the link e_j is

$$S_{ij}(\{p_{rj}\}_{r\in N_j}) = M_{ij}(p_{ij}, \{p_{rj}\}_{r\in N_j\setminus\{i\}})d(e_j).$$

Let x_{ij} be the resource distribution of player i on the link e_j, i.e.

$$x_{ij} = \begin{cases} 1, & e_j \in E^i, \\ 0, & \text{otherwise.} \end{cases}$$

Player i with m_i units of the resource on graph $G(V, E)$ can attract consumers whose number equals

$$S_i = \sum_{j=1}^{|E|} M_{ij}(p_{ij}, \{p_{rj}\}_{r\in N_j\setminus\{i\}})d(e_j)x_{ij}.$$

The gain of player i on the link e_j is equal to the price for the service multiplied by the share of the consumer's demand

$$h_{ij}(\{p_{rj}\}_{r\in N_j}) = p_{ij}M_{ij}(p_{ij}, \{p_{rj}\}_{r\in N_j\setminus\{i\}})d(e_j).$$

Denote by c_{ij} the costs of player i on the link e_j. The costs are proportional to the number of consumers who use the resource. Thus, the general payoff of player i on the graph $G(V, E)$ is

$$H_i(\{p_r\}_{r\in N}, \{x_r\}_{r\in N}) = \sum_{j=1}^{|E|} \big(h_{ij}(p_{ij}, \{p_{rj}\}_{r\in N_j\setminus\{i\}}) - \\ - c_{ij}S_{ij}(p_{ij}, \{p_{rj}\}_{r\in N_j\setminus\{i\}})\big)x_{ij}, \qquad (1)$$

where p is the profile of prices of all players and x defines the allocation of the resources on the network $E^1 \times \ldots \times E^n$.

The game consists of two steps. First, players form their transportation networks (location problem) and then they announce the prices for their service (pricing problem). The consumers are distributed among the services and the players receive the payoffs $H_1, \ldots, H_n$. The objective of a player is to maximize the payoff. The location problem, firstly, installed by Hotelling [4] as a problem of Nash equilibrium of competitive facilities on a linear market, afterwards was considered in linear variant in the articles of d'Aspremont et al. [3], Kats [5], Bester [1], and in plane market for two firms in the article Mazalov and Sakaguchi [6]. Pricing competition among more than two firms was considered in McFadden [7] where sufficient conditions on the existence of Nash equilibrium in pricing game for any numbers of firms are obtained.

In this paper we derive the equilibrium in this location-pricing game for any number of players on the transportation network.

2 Location Game-Theoretic Model on Graph

Let the market is presented by some transportation network $G(V, E)$. On the market there are n companies. Each company allocates m_i transport units on the links of the network. Thus, the firms form the network of routes $E^1 \times \ldots \times E^n$. The allocation is determined by the vectors $x_i, i = 1, \ldots, n$.

$$x_{ij} \in \{0, 1\}, \quad \sum_{r=1}^{|E|} x_{ir} = m_i.$$

Then, the players simultaneously announce the prices $\{p_i\}_{i \in N}$ in their networks $E^i, i \in N$,

$$p_{ij} \in [0, \infty), \quad e^j \in E^i.$$

In every link of the network $G(V, E)$ it is determined a flow of consumers $d(e_j)$ $(e_j \in G(V, E))$. We suppose that the flow depends of the population size P_1, P_2 in the vertexes of the departure and destination:

$$d(e_j) = \frac{\sqrt{P(v_j^1)P(v_j^2)}}{2}, \quad e_j = (v_j^1, v_j^2).$$

The share of the firm i in the flow on the link e_j depends of the price p_{ij} and the prices of the competitors on this link. We suppose that the distribution of consumers follows the multinomial logit-model [7]. So, the share of the firm i in the flow on the link e_j is

$$M_{ij}(p_{ij}, \{p_{rj}\}_{r \in N_j \setminus \{i\}}) = \frac{e^{a_1 p_{ij} + (a, k_{ij})}}{\sum_{s=1}^{|N_j|} e^{a_1 p_{sj} + (a, k_{sj})} + e^{\rho}}, \quad e_j \in E^i,$$

where $a_1 < 0$, a—constant vector, k_{ij} corresponds to route e_j, N_j—number of competitors on the link e_j. The term e^{ρ} corresponds to the part of consumers who are not in the service.

The gain of the firm on the link e_j is equal to

$$h_{ij}(\{p_{rj}\}_{r\in N_j}) = (p_{ij} - c_{ij})M_{ij}d(e_j), \quad i \in N_j.$$

and the general gain is

$$H_i(\{p_r\}_{r\in N}, \{x_r\}_{r\in N}) = \sum_{j=1}^{|E|} h_{ij}(\{p_{rj}\}_{r\in N_j})x_{ij}.$$

We determined n-person non-cooperative game on the set of the strategies (x_i, p_i), $i \in N$.

3 Equilibrium in Location-Pricing Game

Suppose that the players fixed the allocation of the resources x and announce the prices p. In the pricing game the gain of i-th player on the link e_j depends of the profile of prices p_{ij} on this link. So, we can consider the pricing game in each link of the network $G(V, E)$. The existence and uniqueness of the equilibrium was proven in the article [2].

The equilibrium $\{p^*_{ij}\}_{i\in N_j}$ can be constructed as a limit of the sequence of best response strategies. The best response strategy of the player i is satisfied to the equation

$$\big(1 - M_{ij}(p_{ij}, \{p_{rj}\}_{r\in N_j\setminus\{i\}})\big)(c_{ij} - p_{ij}) = \frac{1}{a_1}.$$

We prove that if we introduce a new firm in the pricing game then the payoffs of the players on the link e_j are decreasing.

In the location game for two players we apply the following procedure. Let one of the firms allocates the resources in the network $G(V, E)$. We allocate the resource units of other firm sequentially, one by one, every time finding the equilibrium in prices. The equilibrium we find using the best response strategies. Using the fact that increasing in the number of firms on the link involves the decreasing of the payoffs it is not difficult to show that this sequence of best response strategies converges.

4 Modelling

The model proposed earlier was applied to the model of competition on the Russian and Chinese airline markets. Transportation networks of these markets are presented in Figs. 1 and 2.

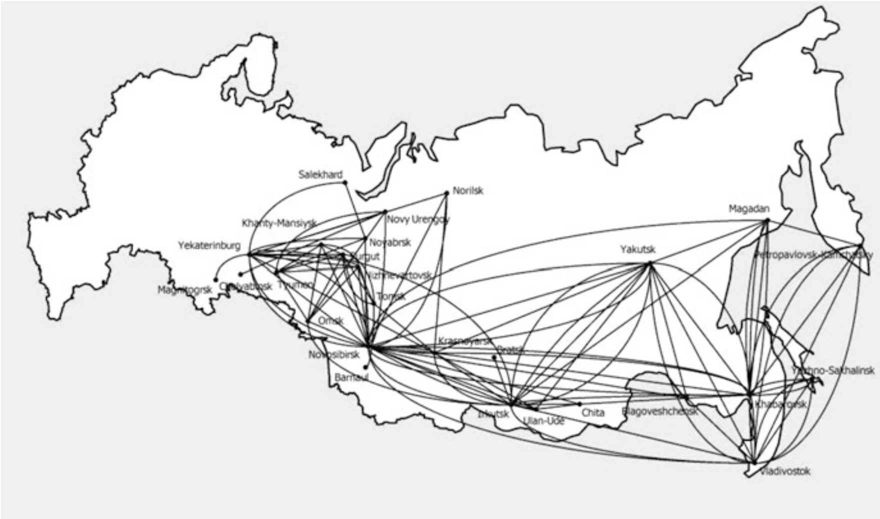


Fig. 1 Russian airline market

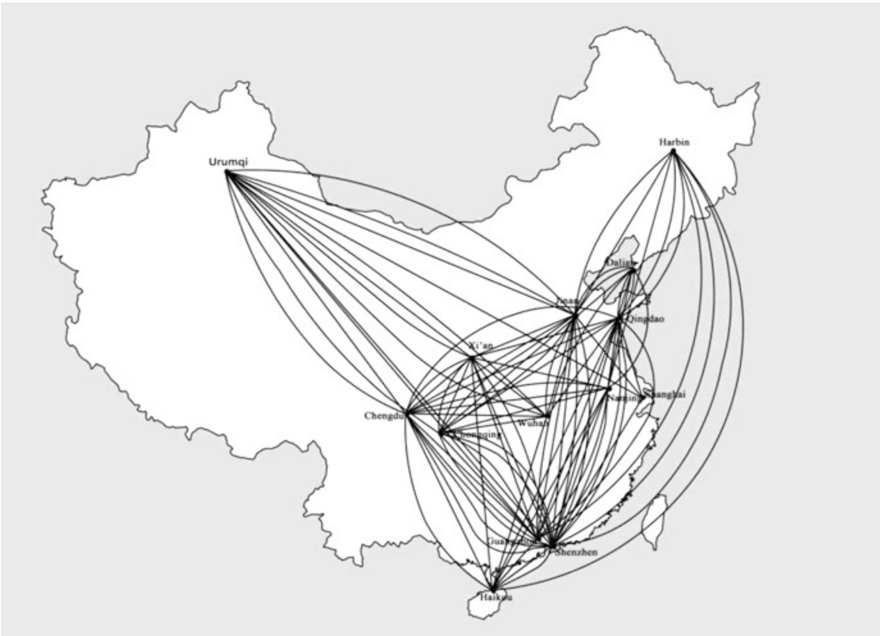


Fig. 2 Chinese airline market

In Table 1 you can see the main indicators of these markets. We see that the number of vertexes and links in the transportation graph of Russian market is larger than in Chinese market. However, the number of flights in Chinese market is larger than in Russian market. So, the level of competition in Chinese market is higher.

Table 1 Market indicators

Indicator	Russia	China
Number of airports	27	14
Number of routes	95	61
Number of direct routes	239	351
Number of non-direct routes	74	14
Number of aircompanies	11	5
Maximal number of aircompanies on the link	5	3
Frequency of flights per week	2.8	6.4

Table 2 Equilibrium on the route Irkutsk-Novosinirsk

Aircompany	Time (h)	Frequence (per week)	Distance (km)	Eq. price (and real price)	Share of market
Siberia (S7)	2.4	4	1462.6	3029.95 (9930)	0.23
IrAero	3.55	5	1520.918	2986.04 (10,930)	0.1
Angara	2.1	3	1462.6	3347.28 (6630)	0.2
Rusline	2.4	3	1462.6	3115.01 (9825)	0.21
NordStar	5.2	3	1520.918	2854.08 (7495)	0.07

Table 3 Equilibrium on the route Nankin-Harbin

Aircompany	Time (h)	Frequence (per week)	Distance (km)	Eq. price (and real price)	Share of market
Shenzhen airlines	2.4	7	1665	709.95 (1650)	0.28
Sichuan airlines	2.5	7	1665	702.96 (1650)	0.28
Xiamen airlines	2.4	6	1665	576.84 (1620)	0.12

In Table 2 the results of calculations for the route Irkutsk-Novosibirsk in the Russian market are presented. In this route five aircompanies make the service. You can compare the equilibrium prices with real prices on the market. There is some disproportion in data. Some companies are supported by the local government. In Table 3 the same values are computed for the Chinese market on the route Nankin-Harbin. There is good correspondence between equilibrium and real prices in the market.

5 Conclusion

This paper has introduced the model of competition of n firms on the transportation network. We present the algorithm to find the equilibrium in this game. For some segments of Russian and Chinese airline market the equilibrium in pricing and location models is derived. Future work could investigate the properties of equilibria under the inclusion to the model some additional factors such as the size of hubs, seat capacity (see [8]), possibility of coalition forming, etc.

Acknowledgements The research was supported by the grant of the Russian Fund for Basic Research (project 13-01-91158) and the Division of Mathematical Sciences of the Russian Academy of Sciences.

References

1. Bester, H., De Palma, A., Leininger, W., Thomas, J., Von Tadden, E.L.: A non-cooperative analysis of hotelling's location game. Games Econ. Behav. **12**, 165–186 (1996)
2. Caplin, A., Nalebuff, B.: Aggregation and imperfect competition: on the existence of equilibrium. Econometrica **59**(1), 25–59 (1991)
3. d'Aspremont, C., Gabszewicz, J., Thisse, J.F.: On hotelling's "stability in competition". Econometrica **47**(5), 1145–1150 (1979)
4. Hotelling, H.: Stability in competition. Econ. J. **39**, 41–57 (1929)
5. Kats, A.: Location-price equilibria in a spatial model of discriminatory pricing. Econ. Lett. **25**, 105–109 (1987)
6. Mazalov, V.V., Sakaguchi, M.: Location game on the plane. Int. Game Theory Rev. **5**, 13–25 (2003)
7. McFadden, D.: Conditional logit analysis of qualitative choice behavior. In: Zarembka, P. (ed.) Frontiers in Econometrics, pp. 105–142. Academic, New York (1973)
8. Swan, W., Adler, N.: Aircraft trip cost parameters: A function of stage length and seat capacity. Transp. Res. E **42**, 105–115 (2006)

A GRASP Algorithm for the Vehicle-Reservation Assignment Problem

Beatriz Brito Oliveira, Maria Antónia Carravilla, and José Fernando Oliveira

1 Introduction

The car rental business is becoming heavily dependent on operational efficiency. As holding costs of assets have been growing faster than the price level, it is important to assure optimal utilization of resources as well as a high service level [3]. The work described in this paper proceeds from a project funded by a Portuguese car rental company whose main objective was to redesign and enhance the company's procedures to assign special types of vehicles to reservations. Due to their unique characteristics, these vehicles are not highly required. Consequently, the number of cars available of each of these special groups is small, thus forcing the company to transfer them empty between rental stations in order to meet the reservations requirements. The company felt the need to improve the assignment process so as to minimize these empty transfers. This paper presents a metaheuristic for this problem. The main objective is to maximize the total profit of the company and assure customer satisfaction by fulfilling as many reservations as possible, whilst reducing the cost of the empty transfers. In fact, the reduction of the empty repositions takes a significant role both in increasing operation efficiency and thus profitability of the business model and in improving the environmental sustainability of the company.

The empty repositions or "deadheading" trips are a studied issue within the transportation industry framework. Dejax and Crainic [1] recognize the impact empty flows have on logistic systems, namely on their operational and economic performance, as they generate costs and no revenue.

B.B. Oliveira • M.A. Carravilla (✉) • J.F. Oliveira
INESC TEC, Faculty of Engineering, University of Porto, 4200-465 Porto, Portugal
e-mail: beatrizbritooliveira@gmail.com; mac@fe.up.pt; jfo@fe.up.pt

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_9

The car rental logistics short-term problem has not been frequently addressed on the transportation logistics literature. Nevertheless, some important contributions can be found. Pachon et al. [4] structure the fleet planning process of a car rental company in three sequential phases: pool segmentation, strategic fleet planning, and tactical fleet planning. The first phase consists on clustering the rental locations of the car rental company in geographically and demand-correlated pools; the different rental stations within a pool share the same fleet, whose number of vehicles is determined in the second phase. The third phase consists on determining the number of vehicles that should be available at each station, in each period of time. The problem of empty transfers is herein considered. Pachon et al. [5] model these three phases and propose solution methodologies considering the hierarchical structure of the decision-making process. None of these works addresses the specific assignment of reservations to the available vehicles. Nevertheless, this could be considered a lower-level (more operational) sub-problem of the tactical fleet planning. To the best of our knowledge, within the car rental logistics optimization framework, the lower-level vehicle-reservations assignment problem has not yet been approached.

The remainder of this paper is organized as follows: the next section briefly describes the problem tackled. The following section presents the solution methods and, finally, the main results, based on real instances, are presented and conclusions are drawn.

2 Problem Description

The problem described in this paper aims to allocate a certain set of reservations to the available special vehicles.

Each reservation has the following characteristics: the date and station in which the customer wants to pick up the vehicle, the date and stations in which the customer wants to deliver it, and the revenue of the reservation. The vehicles are characterized by their current occupation; as they are currently fulfilling a certain reservation, each vehicle will be available when and where that reservation in progress ends. Other parameters of this problem are the costs and the time of the empty transfers from one station to another (Fig. 1). The objective is to maximize the revenue of the assigned reservations deprived of the costs of the empty transfers. The main restrictions of the problem are related to the availability of the vehicles on the moment and location considered.

Since the customers have specific requirements, the company should provide them with exactly what was requested. Yet when that is not possible, it is a common practice in this sector to offer the customer a vehicle from a better group for the same value (upgrade). When that option is not available, the company offers the customer the possibility to rent a vehicle from a worst group with a price discount (downgrade). The possibility to upgrade or downgrade the reservations is beneficial for the company as far as service level is concerned, although it may lead to the fulfilment of reservations for a minor profit.

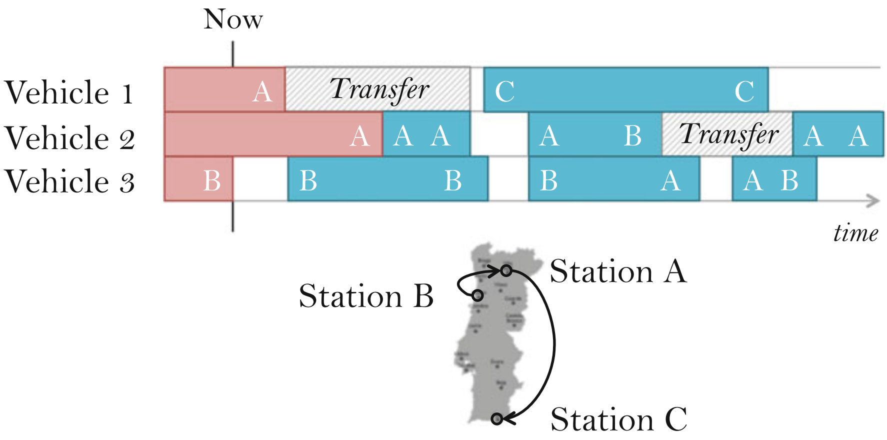


Fig. 1 A vehicle global schedule consists on the scheduling of the reservations fulfilled by each vehicle, which may lead to empty transfer movements

The inclusion of this issue in the solving method increases the dimension of the analysis (as more vehicles are available for the same number of reservations), introducing additional restrictions as for which groups can be upgraded or downgraded to which groups. Hence, reservations need to be characterized by the group (type of vehicle) required by the customer and vehicles should be associated with a specific group. As for the cost of these decisions, in fact, the company does not incur on any additional cost by allocating a better or worse vehicle than requested. Since the vehicles are available, it is better to seize the possible profit of the reservation than not fulfilling it and thus dissatisfying a customer. Nevertheless, it is commonly a company's policy to avoid upgrading a reservation when not necessary and to execute downgrades only if no other option is possible. Both adjustments require the specific authorization of the customer; nevertheless, the upgrade is virtually always accepted.

It is also important to know the status of the reservation, as far as confirmation to the customer is concerned. As to control customer satisfaction, the company is interested in preferring the allocation of confirmed reservations over non-confirmed ones.

3 Problem Solution

The main output expected from this tool is a detailed vehicle global schedule with the reservation assignment plans for each car, as it is shown in Fig. 1 for three example vehicles.

The company felt it was important to obtain good allocation plans. Nevertheless, it was known that these plans, good for the time being, would probably cease to be so due to the dynamic characteristics of the problem. Heuristics were hence drawn to solve the problem due to the need of quickly obtaining a good solution.

The solution to this problem was also based on the need to recognize and model the two functioning modes of the decision support system. On the one hand, the company needs to be able to rapidly establish whether a certain reservation may or not be allocated, considering the current vehicle schedule, in order to promptly answer the client on the phone, confirming or not such reservation—the *online mode*. On the other hand, a more robust method is needed to improve the vehicle schedule built during the day—the *batch mode*. The latter must consider that every confirmed reservation should be necessarily allocated to some vehicle.

A GRASP algorithm (Sect. 3.2) was drawn to fulfil the *batch mode*. Due to the inherent dynamic nature of this problem, it is difficult for the company to register the actual global vehicle schedule for each instance. Consequently, in order to enable the quantification of the improvement brought by the developed support system, a heuristic that mimicked the *online* decision process of the employees in charge of this process (Sect. 3.1) was also developed.

3.1 Mimicry Heuristic

Considering the functioning modes, it is possible to classify the existing procedure in the company as an *online-only* mode. In fact, the employees attempt to allocate each reservation individually, not being able to enhance the global vehicle schedule as a batch. Thus, it is possible to determine that the main inputs for this decision are the single reservation to be allocated and the global vehicle schedule.

The decision process currently followed also depends on the perception that, when there are numerous possible vehicles in which to allocate a certain reservation, the employee may not be able to visually apprehend the occupation of all of them, being limited by the size of the computer screen. Therefore, another important input is the number of vehicles the employee can visualize at the same time and the natural human restrictions when dealing with complex combinatorial problems.

During the decision process, the vehicles are presented in alphabetic order of their license plate. The employee visualizes each vehicle occupation, knowing the starting and returning stations and dates of each reservation allocated to it. The first attempt is to allocate the reservation as the last assignment of one of the visible vehicles, in the best possible position. As there is no information regarding the reservation profit, there is an attempt to reduce the costs by avoiding empty transfers and using as first criterion the coincidence between the returning station of the last reservation and the starting station of the reservation to allocate. The second criterion used is the idle time of the vehicle caused by the hypothetical allocation. The third criterion is to allocate the reservation in the first possible fit between already allocated reservations.

When this procedure is not able to find a feasible solution, a simple switch move within the allocated reservations is tried. For each vehicle, it is verified whether the reservation to be allocated overlaps only one other reservation. If this is the case, it may be possible to allocate the overlapped reservation to another vehicle and insert the unassigned reservation in its previous place. If the feasibility conditions are fulfilled, the simple switch is completed in the first possible fit.

All the previously discussed procedures relate to the visible vehicles on screen. The last alternative procedure is the allocation as the last assignment of a vehicle outside the visible screen; as employees scroll down, if a possible fit is found the reservation is allocated.

3.2 GRASP

The methodology selected to solve this problem, as the *batch* mode is concerned, was GRASP (Greedy Randomized Adaptive Search Procedures), first introduced by Feo and Resende [2]. GRASP is an iterative technique with two sequential phases: the construction of a solution based on a randomized greedy heuristic, and the local search, which applies small adjustments to the solution provided by the first phase with the goal of achieving improvement. Each GRASP iteration comprises these two phases and originates a feasible solution; throughout the iterations, the best solution found is preserved. This methodology was chosen due to its intuitive structure and relatively simple implementation.

The constructive heuristic, which aims to assign reservations to vehicles, is based on the ranking of both reservations and vehicles. The reservation rank is based primarily on its confirmation status followed by the proximity of the starting date of the reservation and finally, by decreasing profit. Subsequently, for each reservation, starting with the one ranking higher, the vehicles which could be available on the required date and location are ranked from the lowest to the highest transfer cost between the returning station of their current reservation and the starting station of the reservation considered. If there is a tie, the vehicles are ranked in ascending order of idle time—the time span between being available in the required location and the start of the reservation.

Two different approaches were designed and tested for the local search routine. The move that defines the neighbourhood structure is based on the swap of pairs of allocated reservations. The selection of the new incumbent solution is different: in the first case the new incumbent solution is the best found in the whole neighbourhood, a best-improvement approach, whilst the second may be described as a first-improvement approach.

The best-improvement approach generates an all-encompassing neighbourhood structure based on a *LPS—List of Possible Swaps* that stores all the possible swaps within the incumbent solution. Each neighbour embodies the incumbent solution modified by one specific swap of the *LPS*. In order to choose the best possible improvement, all neighbours are evaluated. If some improvement in the objective function is possible, the best neighbour becomes the incumbent solution and a new *LPS* is constructed based on it. A new iteration is run and the algorithm stops when no neighbour is able to improve the objective function.

In the first-improvement approach, the local search is also initialized with the selection and listing of the possible swapping pairs within the initial solution (the *LPS*). Each listed pair originates a neighbour—the incumbent solution modified by the swap. The neighbours are only explored until one is found that improves the objective function. In fact, the listed pairs are swapped within each best neighbour that is found and when this happens a new neighbourhood structure is generated. Nevertheless, unlike the previous approach, the algorithm continues to try to swap the pairs listed on the first *LPS* but now within this solution that is now the neighbourhood center. Note that since the listed swaps were selected within a different solution a new feasibility check must be run. Once again, the first neighbour that is able to improve the objective function is selected as the base solution for the neighbourhood generation. The procedure described is repeated until all swaps in *LPS* have been attempted. When an *LPS* has been completely explored, and while it is possible to achieve an improvement, a new *LPS* is generated from the incumbent solution and the process is repeated. It is important to understand that this approach was developed with the objective of obtaining a good, swift routine, which explored the neighbourhood structures in depth rather than in width.

4 Computational Tests and Results

The data used to test this approach was retrieved from the company's database in July, before the beginning of the highest season in the car rental business in Portugal. Therefore, the instances reflect the busiest and most demanding time period faced by the company as far as tactical planning is concerned.

Three instances were selected, each concerning a different vehicle group within the special vehicles fleet and could be classified comparatively as easy (A), average (B), and difficult (C), for both upgrade/downgrade-allowing and -not-allowing situations. These instances were solved using the three routines: the mimicry heuristic was used as a means to simulate the results currently obtained by the company; then, the GRASP algorithm was run using the first-improvement local search as well as the best-improvement one.

Considering the two variations of each of the three instances (allowing for upgrade/downgrade vehicles or not) and the two approaches to the swap local search, twelve different GRASP variants were tested. The algorithms described were developed in a VBA platform, using Microsoft Office Excel as the input/output interface. Each GRASP variant was run for 15 iterations using a standard personal computer with an INTEL i7 2.70 GHz CPU and 8 GB installed memory.

The general results may be found in Table 1, as far as the improvement between the results of the mimicry heuristic and the GRASP algorithm is concerned.

Table 1 Results for the different GRASP variants—percentage of profit improvement and increase in the number of reservations fulfilled when compared to the company's procedures, and global run time

			As for GRASP iterations				Global	
			Average (%)	Std dev (%)	Worst (%)	Best (%)	Time (min)	New reserv
A	No Up	BI	0.1	0.2	−0.2	0.5	6	0
A	No Up	FI	0.0	0.2	−0.3	0.5	1	0
A	Up	BI	−0.1	0.2	−0.4	0.3	59	0
A	Up	FI	−0.1	0.2	−0.5	0.3	7	0
B	No Up	BI	10.5	0.4	10.0	10.7	2	23
B	No Up	FI	10.5	0.4	10.0	10.7	1	23
B	Up	BI	5.4	0.1	5.3	5.5	91	23
B	Up	FI	5.4	0.1	5.3	5.5	19	23
C	No Up	BI	8.4	0.1	8.3	8.5	2	0
C	No Up	FI	8.4	0.1	8.3	8.5	1	0
C	Up	BI	12.0	0.1	11.8	12.1	149	12
C	Up	FI	11.9	0.1	11.8	12.1	47	12

Note: *Up* allowing upgrades/downgrades, *No Up* not allowing upgrades/downgrades, *BI* best-improvement, *FI* first-improvement

It was possible to verify that the metaheuristic approach lead to better results when the difficulty of the instance increased. In fact, for the easy instance (A), the increase on the profit of the company when compared to the values obtained by the mimicry heuristic was virtually non-existent, both considering and not considering upgrades. As for the average instance (B), when considering that no upgrades or downgrades to other groups were possible, both local search routines were able to increase the company's profit in 10.7 %. When considering upgrading and downgrading vehicles, there was a 5.5 % increase. The difficult instance (C), solved by both local search routines, whilst not considering upgrades, lead to an increase of over 8.5 % of the profit. When considering these auxiliary vehicles, both routines were able to increase the results of the company by 12.1 %. These increasing values of improvement were expected, as the current procedure used by the company, although extremely refined by the experience and knowledge of the operators, meets the limits of the human ability to apprehend large amounts of data and thus tackle big combinatorial problems.

For every instance and upgrading situation, the swift first-improvement (FI) local search solved the problem faster; in fact, the time was perceived to be proportional to the amount of vehicles to assign, increasing when considering upgrades and when solving instances with more vehicles available. For every case, nevertheless, the algorithms were run in an acceptable time, considering that this *batch* approach to the problem is designed to be run during the night. For most cases, the swift local search was able to match the results of the best-improvement routine. Nevertheless, for instance C, considering upgrades, a worst result was obtained using the first-improvement heuristic than the exhaustive best-improvement approach; nonetheless, this difference represented only 0.02 % of the profit. For the cases considered, the contribution of the local search to the overall improvement was between 0.5 and 2 %, decreasing with the difficulty of the instance.

It should also be noticed that for the average instance (B) and for the upgrade-allowing difficult instance (C), this approach was also able to allocate new reservations that the mimicry heuristic was not able to insert in the global vehicle schedule.

5 Conclusions

This paper reported the problem faced by a car rental company in managing the tactical planning of its special vehicles fleet, as far as allocation of reservations to available vehicles is concerned. The main objective was to maximize the total profit of the company, by means of reducing empty vehicle repositioning transfers between rental stations. The ultimate goal was thus to provide the company with a functioning tool that was able to improve the global vehicle schedule considering the required rental groups that belonged to the *special vehicle* fleet.

In the system developed, the user is able to select the rental groups to improve, allow upgrades and/or downgrades, and select the local search approach, considering the time available to run it. The outputs of each run consist on a detailed schedule for the vehicles of each group and a chronogram with the vehicle occupation.

For the most difficult instances and groups, the company is able to increase its profit up to 12 %. It is significant to remark that this value, referent to a volume of reservations that historically comprehends most of the volume of the high season demand, represents a significant financial impact for the company. Moreover, the company may also be able to fulfil more reservations, increasing customer satisfaction and market share. Another major advantage is the re-allocation of two qualified and experienced employees to other value-adding tasks, namely within the strategic rather than tactical planning level. In fact, this software may also provide the company with insights related to the strategic fleet sizing problem, as it can be used as a simulation tool for the sale and purchase of vehicles.

Nevertheless, it is still possible to improve the approach to this problem. One of the main characteristics of this problem is the extreme inflexibility of the starting and finishing times of the reservations. If there are many reservations concentrated in a specific time period, the problem becomes even more rigid and the solutions more difficult to improve by this method, since small adjustments made to a specific solution lead often to infeasible results. In fact, for such cases, it is the randomness applied to the solving method that leads to the major improvement. Therefore, a new approach could be attempted, namely solving this combinatorial problem by an optimization method.

As future work, it would be also interesting to tackle the issue raised by Sbihi and Eglese [6], related to the measuring of the environmental impacts; the authors state that travel time is a better estimate to degree of pollution caused, as this can be reduced by travelling for shorter times (and at better speeds). Therefore, it could be interesting to reformulate the problem as to minimize the empty transfer times, rather than the costs.

Acknowledgements The authors would like to thank the anonymous reviewers for their valuable comments that helped to improve the paper.

This work was partially funded under project "Smart Manufacturing and Logistics—NORTE-07-0124-FEDER-000057" by the North Portugal Regional Operational Programme (ON.2—O Novo Norte), under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF), and by national funds, through the Portuguese funding agency, Fundação para a Ciência e a Tecnologia (FCT).

References

1. Dejax, P.J., Crainic, T.G.: A review of empty flows and fleet management models in freight transportation. Transp. Sci. **21**(4), 227–247 (1987)
2. Feo, T.A., Resende, M.G.C.: Greedy randomized adaptive search procedures. J. Glob. Optim. **6**, 109–133 (1995)
3. Fink, A., Reiners, T.: Modeling and solving the short-term car rental logistics problem. Transp. Res. E Logist. Transp. Rev. **42**(4), 272–292 (2006)
4. Pachon, J.E., Iakovou, E., Ip, C., Aboudi, R.: A synthesis of tactical fleet planning models for the car rental industry. IEE Trans. **35**(9), 907–916 (2003)
5. Pachon, J., Iakovou, E., Ip, C.: Vehicle fleet planning in the car rental industry. J. Revenue Pricing Manag. **5**(3), 221–236 (2006)
6. Sbihi, A., Eglese, R.W.: Combinatorial optimization and green logistics. 4OR **5**(2), 99–116 (2007)

A Heuristic for the Time-Dependent Vehicle Routing Problem with Time Windows

Vincent Huart, Sylvain Perron, Gilles Caporossi, and Christophe Duhamel

1 Introduction

Routing problems are often the core component of transportation systems in logistics. The Vehicle Routing Problem (VRP) is the most prominent and it consists in building a minimal cost set of routes to visit a set of clients with known demands. The VRP has been extensively studied both with exact and heuristic methods (see the surveys [4, 10]). We consider the Time-Dependent Vehicle Routing Problem with Time Windows (TDVRPTW), which extends the VRP with time windows and time-dependent travel duration. Thus, given a digraph with a speed profile on each arc (a staircase function depending on the time) and a time window for each node, the TDVRPTW consists in building a minimal set of routes visiting each client once, satisfying the time window constraints and the vehicle capacity constraint. This problem is NP-hard as it generalizes the VRPTW.

In the TDVRP, the travel time between two clients can vary over the time. Its main interest is to better handle the dynamic nature of the traffic, especially in urban contexts where there are often strong variations. By doing so, one can expect finding routes that avoid congested areas when traffic jam occurs.

Malandraki and Daskin [9] propose a model and a heuristic for the TDVRP. Hill and Benton [6] focus on the way to represent and store data, which is of practical importance since the shortest path between two nodes may change over the time. Ichoua et al. [7] present a tabu search for the TDVRPTW and they introduce the First

V. Huart • S. Perron • G. Caporossi
GERAD and HEC Montréal, Montréal, QC, Canada
e-mail: vincent.huart@hec.ca; sylvain.perron@hec.ca; gilles.caporossi@hec.ca

C. Duhamel (✉)
LIMOS, Université Blaise Pascal, Clermont-Ferrand, France
e-mail: christophe.duhamel@isima.fr

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_10

In First Out (FIFO) property: a vehicle v_2 entering an arc (i, j) after a vehicle v_1 cannot arrive at j before v_1. Thus, no overtake is allowed in the traffic and delaying a departure cannot improve the routing. This assumption is kept in our model. Several other metaheuristics have been proposed since (genetic algorithm [5], ant colony algorithm [3], simulated annealing [8]).

The proposed column generation heuristic for solving the TDVRPTW is presented in Sect. 2. Computational experiments follow in Sect. 3, before concluding remarks.

2 Heuristic

Given a speed profile for each arc, the first step consists in computing and storing the travel duration between any pair of nodes and for any moment of the time period. A modified Dijkstra algorithm is used and the resulting shortest paths are stored in a map, along with the associated time interval.

Our heuristic relies on the Column Generation (CG) technique. Desrochers et al. [2] present a set partitioning formulation of the VRPTW and develop an exact CG algorithm where the subproblem (SP) is formulated as a shortest path with time windows. It is solved by dynamic programming. The same master problem (MP) formulation may be used for the TDVRPTW but the subproblem is more difficult to solve since it involves time-dependency. More specifically, the MP is formulated as follows:

$$minimize \sum_{i \in P} c_i x_i \tag{1}$$

$$subject\ to$$

$$\sum_{i \in P} v_{i,j} x_i = 1, \forall j \in V \tag{2}$$

$$x_i, \in \{0, 1\}, \forall i \in P \tag{3}$$

where V is the set of clients and P denotes the pool of columns (route patterns) already added to the MP; binary variable x_i indicates if pattern i is selected or not; constant $v_{i,j}$ states if client $j \in V$ belongs to the route $i \in P$; c_i is the cost of column i. It can be set to the duration of the route if the objective is to minimize the total duration or to 1 if the objective is to minimize the total number of vehicles.

Partitioning constraints (2) may be relaxed:

$$\sum_{i \in P} v_{i,j} x_i \geq 1, \forall j \in V \tag{4}$$

In order to get pricing (dual) information for the SP, integer constraints have to be relaxed and therefore constraints (3) are replaced by:

$$0 \leq x_i, \forall i \in P \tag{5}$$

The role of the subproblem is to generate new columns (routes patterns) to be added to the MP. A column $i \notin P$ is added if its reduced cost $\overline{c_i}$ is negative. Using the notation introduced earlier, the reduced cost of column i of the *relaxed* MP may be computed by:

$$\overline{c_i} = c_i - \sum_{j \in V} v_{i,j} \pi_j \tag{6}$$

where π_j is the dual variable associated to the covering constraint (4) of client j. Replacing each parameter $v_{i,j}$ by a binary decision variable y_j equal to 1 if client j is part of the route and 0 otherwise, we obtain the following optimization problem:

$$\min_{y \in Y} f(y) - \sum_{j \in V} \pi_j y_j \tag{7}$$

where Y is the set of routes satisfying constraints on the vehicle capacity, the time-windows for the clients, and the duration computed according to time-dependency and $f(y)$ is the cost function of the MP for the column associated to y.

We propose to solve it heuristically in the following way: columns (routes) are stored in a pool shared by all the components of the method. Once the relaxed master problem has been solved by Cplex, the duals variables are first extracted. Local searches then select randomly routes from the pool and iteratively apply moves in order to improve the reduced cost. If the reduced cost of the resulting routes is negative, they are inserted into the pool. They are dropped otherwise.

Each local search relies on a single neighborhood structure. A set of seven neighborhood structures is used: (1) *internal 2-opt*, two clients in the same route are swapped; (2) *internal exchange*, generalizing the previous one up to four clients; (3) *external exchange*, two clients in two different routes are swapped; (4) *cross*, two routes are split and the first part of each one is connected to the second part of the other one; (5) *reorganization*, optimizing the clients sequence in a route rebuild from scratch; (6) *internal relocate*, a special case of the previous neighborhood structure limited to two clients; (7) *external relocate*, up to two clients from a route are transferred into an another route. Thus 12 local searches are defined.

Another column generator is proposed: MIP + VND. It consists in solving the integer master problem using all the columns in the pool. The optimal solution is then improved by a Variable Neighborhood Descend (VND). Therefore, the MIP + VND has two roles: (1) find an integer solution to the global problem by solving a MIP and (2) improve this solution by generating new routes using a VND heuristic. The VND method has been introduced by Brimberg [1] and it consists in sequentially using neighborhood structures to improve the solution. Those structures are usually sorted according to their complexity. If an improving solution is found, the current solution is updated and the search resumes to the first structure. Otherwise, the current solution is kept unchanged and the search considers the next structure. The search stops when the last structure fails to generate an improving solution and the current solution is then optimal with respect to all the structures. In our

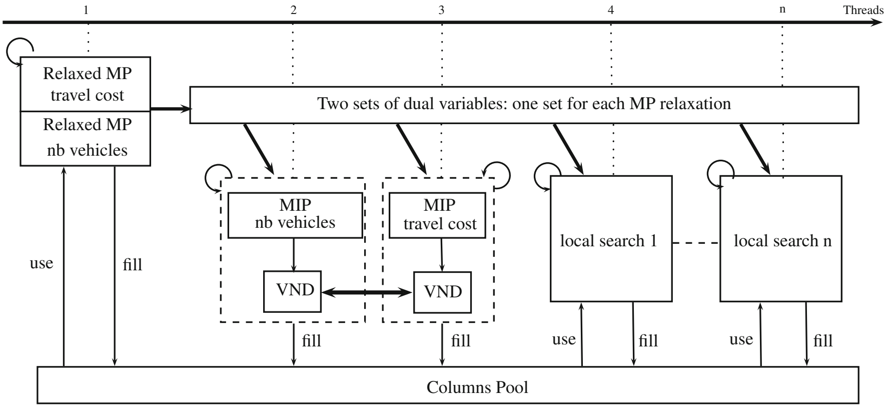


Fig. 1 Global design of the parallel column generation heuristic

implementation, all the neighborhood structures previously mentioned are used. The columns corresponding to the solution are then inserted into the pool.

Moreover, the method considers simultaneously two objectives: the minimum number of vehicles and the minimum travel cost. This leads to 2 MIP+VND generators as well as two restricted master problems. The two sets of dual variables are simultaneously handled in each local search and the general schema of our heuristic is shown in Fig. 1.

The pool is initially filled by three heuristics: (a) a greedy insertion heuristic (b) routes containing a single client and (c) a heuristic which randomly fills the routes. The column generation process iterates until both MIP+VND generators fail at finding improving solutions.

3 Preliminary Results

The computational experiments are done on an Intel Xeon E5-2665 2.40 GHz processor. One thread is dedicated to the MP, two threads for the VND and the remaining 12 threads provide diversity with different neighborhoods. Results are reported for an instance based on the C101 Solomon's instance with 25 clients. Some edges have been removed and a speed profile has been assigned to each arc. The following five different scenarios are defined (see Table 1), time being in minutes from 0:00 am: (0) constant speed profile, original instance, (1) edge differentiation, slow/moderate/fast speed, (2) slowdowns at rush hours on profiles 2 and 3, (3) same as (2) with jams on profile 2 and (4) slowdowns on profile 2 and jams on profile 3.

Table 2 reports the results for the instance with those speed profiles. Both optimization criteria are considered. As can be expected, the travel duration decreases

Table 1 Scenarios for the speed profiles

		Speed inside time intervals in min						
Scenario	Profile	[0–420]	[421–540]	[541–720]	[721–840]	[841–1020]	[1021–1140]	[1141–1440]
0	1	1						
	2							
	3							
1	1	1						
	2	1.8						
	3	2.5						
2	1	1	1	1	1	1	1	1
	2	1.8	1	1.5	1.3	1.5	1	1.8
	3	2.5	1.5	2	2.2	2	1.5	2.5
3	1	1	1	1	1	1	1	1
	2	1.8	0.5	1.5	0.5	1.5	0.5	1.8
	3	2.5	1.5	2	2.2	2	1.5	2.5
4	1	1	1	1	1	1	1	1
	2	1.8	1	1.5	1.3	1.5	1	1.8
	3	2.5	0.5	2	1	2	0.5	2.5

Table 2 Results on C101 with 25 clients

	Scenario 0	Scenario 1	Scenario 2	Scenario 3	Scenario 4
	Minimizing the nb. vehicles				
Duration	2793.9	2410.2	2502.3	2559.8	2904.2
Distance	294.6	318.6	295.5	383.2	404.2
Waiting time	249.3	0.0	82.9	90.7	310.2
Nb. vehicles	3	3	3	3	3
	Minimizing the routes duration				
Duration	2496.2	2383.0	2396.2	2447.4	2904.2
Distance	246.2	250.4	248.3	294.5	404.2
Waiting time	0.0	0.0	0.0	0.8	310.2
Nb. vehicles	4	4	4	5	3

when the speed increases. When optimizing the route duration, the waiting time is also reduced. This may explain the intuitive result that some vehicles can be used twice a day, staying at the depot in the middle of the day. Those vehicles will not be jammed in the traffic when high congestion occurs. Thus, for scenario 2 the waiting time would be 90.0 if such break is not allowed. Splitting the day in two (morning and afternoon) allows better results. In scenario 4, such a split cannot be done because all arcs leaving the depot belong to profile 3.

4 Conclusion

We have considered an extension of the VRPTW in which the travel time changes over the day. Using time-dependency to model the real network is important as it corresponds to real-life urban situation. The method we propose combines a metaheuristic (with VND) with column generation in a multi-threaded framework. The preliminary results on a instance with 25 clients show our approach is promising. It will now be tested on larger instances and several variations will be considered.

References

1. Brimberg, J., Hansen, P., Mladenovic, N., Taillard, E.D.: Improvements and comparison of heuristics for solving the multisource weber problem. Oper. Res. **48**, 444–460 (1997)
2. Desrochers, M., Desrosiers, J., Solomon, M.: A new optimization algorithm for the vehicle routing problem with time windows. Oper. Res. **40**(2), 342–354 (1992)
3. Donati, A.V., Gambardella, L.M., Rizzoli, A.E., Casagrande, N., Montemanni, R.: Time dependent vehicle routing problem with an ant colony system. Eur. J. Oper. Res. **185**, 1174–1191 (2008)
4. Golden, B., Raghavan, S., Wasil, E. (eds.): Vehicle Routing Problem: Latest Advances and New Challenges. Operations Research Computer Science Interfaces, vol. 43. Springer, New York (2008)
5. Haghani, A., Jung, S.: A dynamic vehicle routing problem with time-dependent travel times. Comput. Oper. Res. **32**, 2959–2986 (2005)
6. Hill, A.V., Benton, W.C.: Modeling intra-city time-dependent travel speeds for vehicle scheduling problems. J. Oper. Res. Soc. **43**, 343–351 (1992)
7. Ichoua, S., Gendreau, M., Potvin, J.Y.: Vehicle dispatching with time-dependent travel times. Eur. J. Oper. Res. **144**, 379–396 (2003)
8. Kuo, Y.: Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem. Comput. Ind. Eng. **59**(1), 157–165 (2010)
9. Malandraki, C., Daskin, M.S.: Time dependent vehicle routing problems: formulations, properties and heuristic algorithms. Transp. Sci. **26**, 185–200 (1992)
10. Toth, P., Vigo, D. (eds.): The Vehicle Routing Problem. SIAM Monographs on Discrete Mathematics and Applications, vol. 9. SIAM, Philadelphia, PA (2002)

Integer Programming Based Approaches for Multi-Trip Location Routing

Rita Macedo, Bruna Ramos, Cláudio Alves, José Valério de Carvalho, Saïd Hanafi, and Nenad Mladenović

1 The Multi-Trip Location Routing Problem

The multi-trip location routing problem is a management science problem that occurs typically in the logistics and transportation field. It consists in selecting the depots that should be opened and the corresponding routes to serve a set of clients at minimum cost. The particularity of the multi-trip variant is that a vehicle can now make more than a single route during the planning horizon. The consequence is an increase in complexity since an assignment of routes to vehicles has to be determined.

The problem is characterized by a set $N = \{1, \ldots, n\}$ of clients to visit, whose demand is denoted by b_i, with $i \in N$. The depots have a limited capacity represented by L_d, with $d \in D$, and a fleet of vehicles assigned to them. The fleets of vehicles are assumed to be homogeneous with capacity Q. A vehicle can perform several routes during the planning period. Each route must start and end at the same depot, but a vehicle may perform more than a route per planning period as long as it does not travel more than W units of time (the length of the planning period). We will denote by workday the set of routes assigned to a vehicle. The cost of a solution depends on the fixed costs C_f^d, $d \in D$, of opening a depot d, on the fixed cost C_v incurred each time a vehicle is used, and on the variable costs related to the travelled distances.

R. Macedo • S. Hanafi • N. Mladenović
Université de Valenciennes, 59313 Valenciennes, France
e-mail: Rita.SantosDeMacedo@univ-valenciennes.fr; Said.Hanafi@univ-valenciennes.fr; Nenad.Mladenovic@univ-valenciennes.fr

B. Ramos • C. Alves (✉) • J. Valério de Carvalho
Universidade do Minho, 4710-057 Braga, Portugal
e-mail: bruna.ramos@dps.uminho.pt; claudio@dps.uminho.pt; vc@dps.uminho.pt

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_11

The multi-trip location routing problem has been addressed previously in [1, 3–6]. In [3], Lin et al. explored the problem using heuristics and branch-and-bound. In [4], Lin and Kwok addressed a multi-objective case combining cost minimization with the minimization of the imbalance among vehicles. In [1], the authors propose a compact three-index commodity flow formulation for the problem, and a branch-and-price algorithm based for a column generation reformulation of the problem.

In this paper, we explore a network flow model for the multi-trip location routing problem with no additional constraints imposed to the vehicle routes as opposed to the cases considered in [5, 6]. We describe a stronger formulation, and we present also different valid inequalities to improve the quality of its continuous lower bound. Additionally, we describe an iterative rounding heuristic that relies on this model, and which proved to be effective in finding good incumbents for the problem. To illustrate the efficiency of our approaches, we report on several computational experiments on benchmark instances that we compare with the results obtained using a three-index commodity flow model proposed by Akca et al. in [1].

In Sect. 2, we briefly recall the three-index commodity flow formulation proposed by Akca et al. [1]. The network flow formulation and the corresponding valid inequalities that we derived for it are presented in Sect. 3. The iterative rounding heuristic is described in Sect. 4. In Sect. 5, we present our comparative computational experiments among the different approaches discussed in this paper.

2 A Three-Index Commodity Flow Model

Let G be a graph with a set of nodes associated to the depots and to the clients, and a set of arcs between each pair depot-client and client-client, such that $G = (N \cup D, A)$, with $A = (D \times N) \cup (N \times N) \cup (N \times D)$. The complete set of vehicles is denoted by H, with H_d being the subset of the vehicles assigned to a depot d. The travel time between nodes i and j, with $(i, j) \in A$ is denoted by t_{ij}, while the cost associated to an unit of time is denoted by C^o.

The three-index commodity flow model of Akca et al. [1] has variables related to the opening of the depots and to the vehicles usage and operation. The binary variables λ_d, $d \in D$, state if a depot is selected to be open or not. The usage of a vehicle h is represented by the binary variables v_h, $h \in H$. If a vehicle h goes through an arc $(i, k) \in A$, then the corresponding variable x_{ikh} will take the value 1, and 0 otherwise. The amount of load that the vehicle h carries through (i, k) is denoted by y_{ikh}.

$$\min \sum_{d \in D} C_f^d \lambda_d + C_v \sum_{h \in H} v_h + C^o \sum_{h \in H} \sum_{(i,k) \in A} t_{ik} x_{ikh} \tag{1}$$

$$\text{s.t.} \sum_{h \in H} \sum_{k \in (N \cup D)} x_{ikh} = 1, \quad \forall i \in N, \tag{2}$$

$$\sum_{k \in (N \cup D)} x_{ikh} - \sum_{k \in (N \cup D)} x_{kih} = 0, \quad \forall i \in N \cup D,\ \forall h \in H, \tag{3}$$

$$\sum_{h \in H_d} \sum_{k \in N} y_{dkh} \leq L_d \lambda_d, \quad \forall d \in D, \tag{4}$$

$$y_{ikh} \leq Q x_{ikh},\ \forall (i,k) \in A, \quad \forall h \in H, \tag{5}$$

$$\sum_{k \in N} y_{ikh} - \sum_{k \in N} y_{kih} + b_i \sum_{k \in (N \cup D)} x_{ikh} = 0, \quad \forall i \in N, \forall h \in H, \tag{6}$$

$$\sum_{(i,k) \in A} t_{ik} x_{ikh} \leq W v_h, \quad \forall h \in H, \tag{7}$$

$$x_{dkh} = 0, \quad \forall d \in D, \forall k \in (N \cup D), \forall h \in H_t, \forall t \in D \backslash \{d\}, \tag{8}$$

$$x_{ikh} \in \{0,1\}, \quad \forall (i,k) \in A, \forall h \in H, \tag{9}$$

$$y_{ikh} \geq 0, \quad \forall (i,k) \in A, \forall h \in H, \tag{10}$$

$$\lambda_d \in \{0,1\}, \quad \forall d \in D, \tag{11}$$

$$v_h \in \{0,1\}, \quad \forall h \in H. \tag{12}$$

The mandatory visit to every client is represented by the constraints (2)–(3). The capacity constraints of the depot and vehicles is expressed through constraints (4) and (5)–(6), respectively. Constraints (7) forbid a vehicle to travel more than W units of time, while (8) force a vehicle to travel only through the arcs associated to its depot. The function (1) denotes the total cost minimization objective.

3 Network Flow Formulation

3.1 The Model

Our network model is defined on acyclic directed graphs (one per depot) that we will denote by $\Pi_d = (\Delta, \Psi_d)$, $d \in D$. A path on these graphs correspond to the workday of a given vehicle from its corresponding depot. The vertices in Δ represent discrete time instants starting from 0 up to the time limit W. The arcs are associated to vehicle routes, and additionally to waiting periods at the depot. An arc $(u, v)^r \in \Psi_d$ is related to a route r that starts at time instant u and ends at time instant v. The set Ψ_d is defined as follows: $\Psi_d = \{(u, v)^r : 0 \leq u < v \leq W, r \in R_d\} \cup \{(u, v)^o : 0 \leq u < v \leq W, v = u + 1\}$, with R_d being the set of all the routes from depot d. The load, duration and cost of a route r will be denoted by l_r, t_r and C_r, respectively. The set of clients visited by a route r will be represented by N_r, with $N_r \subseteq N$. Clearly, a route is feasible only if $l_r \leq Q$ and $t_r \leq W$.

The model is composed by two sets of variables. The binary variables λ_d, $d \in D$, state whether a depot is selected to be open or not, while the binary variables x^d_{uvr} state whether the route r associated to depot d, which starts and finishes at time instants u and v respectively, is to be used or not. The network flow formulation is defined as follows.

$$\min \sum_{d \in D} \sum_{(u,v)^r \in \Psi_d} C_r x^d_{uvr} + C_v \sum_{d \in D} \sum_{(0,v)^r \in \Psi_d} x^d_{0vr} + \sum_{d \in D} C^d_f \lambda_d \tag{13}$$

$$\text{s.t.} \sum_{d \in D} \sum_{(u,v)^r \in \Psi_d | i \in N_r} x^d_{uvr} = 1, \quad \forall i \in N, \tag{14}$$

$$\sum_{(0,v)^r \in \Psi_d} x^d_{0vr} \leq K^{max}_d \lambda_d, \quad \forall d \in D, \tag{15}$$

$$- \sum_{(u,v)^r \in \Psi_d} x^d_{uvr} + \sum_{(v,y)^t \in \Psi_d} x^d_{vyt} = \begin{cases} 0, \text{ if } v = 1, \dots, W-1, \\ -\sum_{(0,v)^r \in \Psi_d} x^d_{0vr}, \text{ if } v = W, \end{cases} \quad \forall d \in D, \tag{16}$$

$$\sum_{(u,v)^r \in \Psi_d} l_r x^d_{uvr} \leq L_d \lambda_d, \quad \forall d \in D, \tag{17}$$

$$x^d_{uvr} \geq 0, \quad \text{and integer}, \forall (u,v)^r \in \Psi_d, \ \forall d \in D, \tag{18}$$

$$\lambda_d \in \{0, 1\}, \quad \forall d \in D. \tag{19}$$

Constraints (14) force the visit to every client. Constraints (15) limit the number of workdays per depot to a maximum of K^{max}_d. Note that x^d_{0vr} is directly related to an independent workday starting at time instant 0 from depot d and finishing at time instant v. If the corresponding depot is not selected to be open, the maximum number of workdays becomes naturally 0. Every upper bound on the number of workdays can be used for K^{max}_d. In our experiments, we used the following value (assuming that the clients are ordered in decreasing order of their demands): $K^{max}_d = \max \left\{ j : \sum_{i=1}^{j} b_i \leq L_d \right\}$, for a given depot $d \in D$. Flow conservation is enforced through constraints (16). Constraints (17) ensure that the capacities of the depots are not exceeded. The objective function is represented through the expression (13).

3.2 Valid Inequalities

To improve the quality of the continuous lower bounds obtained with the network flow model (13)–(19), the following valid inequalities can be used. The first consists in forcing a minimum number D^{min} of depots to be open through the following constraint: $\sum_{d \in D} \lambda_d \geq D^{min}$. The problem of determining D^{min} is a one-dimensional bin-packing problem. To compute its value, we resorted to dual-feasible functions [2] which provide a mean to obtain high quality lower bounds frequently near from those achieved with column generation models.

The second inequality is similar to the previous one, but applies now to the vehicles. The principle is to enforce a minimum number H^{min} of vehicles to use through the following constraint: $\sum_{d \in D} \sum_{(u,v)^r \in \Psi_d} x^d_{uvr} \geq H^{min}$. Again, H^{min} is a lower bound for the bin-packing problem defined by using the clients demands and the vehicles capacities, and it can be computed using the aforementioned dual-feasible functions.

The last set of inequalities consists in relating the selection of workdays to the opening of depots. These inequalities state that if a depot is opened, there should be at least one workday to be performed from this depot: $\sum_{(0,v)^r \in \Psi_d} x^d_{0vr} \geq \lambda_d$, $\forall d \in D$.

4 An Iterative Rounding Heuristic

As our computational experiments will show in Sect. 5, the network flow model proved to be very effective for deriving good incumbents for the problem. A fast procedure for obtaining good quality solutions consists in the following rounding heuristic that relies on the iterative solution of the linear relaxation of (13)–(19).

The heuristic starts with the solution of the linear relaxation of (13)–(19). It follows with an attempt to fix the variables of this model starting by the λ_d variables, and then the x^d_{0vr} and the x^d_{uvr} with $u > 0$ in that order and repeatedly. The principle is to force sequentially the opening of the depots whose corresponding λ_d is above a given parameter α (starting by the depot with the largest λ_d and so on), and then to build workdays for the selected depot d first by rounding up a x^d_{0vr} variable whose value is above a given parameter β (again, the x^d_{0vr} variables are selected in decreasing order of their value) and then by selecting further routes $(u, v')^{r'} \in \Psi_d$ (with $u >= v$, and in increasing order of u) to pair with the previous one such that $x^d_{uv'r'} > \beta$ and until there remain routes given the time limit W. Once there are no more variables to fix, the linear relaxation of (13)–(19) is solved again for the remaining instance, and the process repeats until it cannot fix anymore variables. At this stage, if a solution for the original problem is not already available, the model (13)–(19) is solved up to integrality for the remaining instance. In that case, a limit on the computing time can be enforced, and the best incumbent found until this time limit can be used as a solution.

5 Computational Results

In this section, we report on the computational experiments performed on benchmark instances to evaluate the performance of the models discussed in this paper both in terms of the quality of their lower bounds, and on their ability to drive efficiently the search for good quality integer solutions. The tests were run on a PC with an i7 CPU with 3.5 GHz and 32 GB of RAM. The optimization subroutines rely on CPLEX 12.5. For our experiments, we used a set of 40 benchmark instances from the literature whose relevant parameters are given in Tables 1 and 2.

The tests are divided in three parts. First, we compare the quality of the continuous lower bounds of the models (1)–(12) and (13)–(19) without enforcing any other valid inequality. The results of these tests are listed in the first part of Table 1. The columns z_{RL} and t_{RL} denote respectively the value of the lower bound and the computing time (in seconds) required for the solution of the linear programming relaxation of the corresponding model by CPLEX. Column *lb* represents respectively the best lower bound obtained when the corresponding model is solved by CPLEX up to integrality and after a maximum of 900 s of computation. The columns *ub* and t_{UB} denote respectively the value of the best incumbent found within this time limit, and the total execution time in seconds (which is smaller than 900 s only if a proven optimal solution has been found within the time limit). The columns *gap* provide the value in percentage of the optimality gap reached at the end of the solution procedure. A "–" entry denotes the fact that no feasible integer solution was found. Finally, column t_g gives the total computing time required to generate the routes for model (13)–(19). The second set of experiments is reported in the second part of Table 1. The same tests as above were repeated enforcing now the valid inequalities described in Sect. 3.2. For a fair evaluation, and since the first and second cut described in this section can also be enforced in model (1)–(12), we solved again this model using these two cuts. The last set of experiments was performed to evaluate the iterative rounding heuristic described in Sect. 4 considering the model (13)–(19) both with and without cuts. The results are given in Table 2. Column *best ub* gives the value of the best known upper bound for the corresponding instance, columns ub_h represent the value of the solution found by the heuristic, and t_{ub_h} the total computing time in seconds. In these experiments, we limited the total time spent in solving the remaining instance up to integrality to 30 s. The parameters α and β were both set initially to 0.9. If the process fails in fixing variables, then it is repeated with smaller values of α (0.75 and 0.5) and β (0.5 and 0.25). If it still fails after using these values, we resort to the exact solution procedure for the remaining instances as described in Sect. 4.

From the results obtained, it is clear that the continuous lower bound of (13)–(19) is better than the bound of (1)–(12) both with and without enforcing additional inequalities. Furthermore, the network flow formulation (13)–(19) proved to be much more effective than (1)–(12) in finding good quality integer solutions. In some cases, the model of Akca et al. fails even in finding a feasible solution, when (13)–(19) provides an optimal integer solution. For 18 instances, model (13)–

Table 1 Computational results (Part I)

				Model (1)–(12)						Model (13)–(19)						
Inst.	*n*	*W*	*Q*	z_{RL}	t_{RL}	*lb*	*ub*	t_{UB}	*gap*	z_{RL}	t_g	t_{RL}	*lb*	*ub*	t_{UB}	*gap*
1	20	140	50	3151.24	0.62	3359.31	4671	900.15	28.08	3254.44	1.04	0.11	4431.00	4431	29.73	0.00
2	20	160	50	3095.01	0.59	3218.96	4656	900.20	30.86	3192.87	1.19	0.21	4332.58	4430	900.87	2.20
3	20	140	70	3071.06	0.64	3174.38	4618	900.17	31.26	3181.82	51.04	0.24	4335.69	4384	901.11	1.10
4	20	160	70	3017.67	0.64	3069.76	4399	902.73	30.22	3122.97	58.96	0.56	4187.46	4378	901.96	4.35
5	20	140	50	2856.71	0.61	2903.55	4764	900.20	39.05	3059.33	2.22	0.07	4681.00	4681	701.16	0.00
6	20	160	50	2790.07	0.64	2882.60	4780	902.99	39.69	2949.29	2.58	0.13	4472.00	4472	64.06	0.00
7	20	140	70	2760.71	0.64	2800.63	4674	902.24	40.08	2956.06	156.70	0.18	4424.00	4424	188.28	0.00
8	20	160	70	2698.69	0.62	2736.22	4697	900.18	41.75	2873.88	181.02	0.35	4406.04	4424	900.68	0.41
9	20	140	60	3471.53	0.66	3654.39	4684	900.17	21.98	3598.47	0.10	0.10	4431.00	4431	33.00	0.00
10	20	160	60	3417.93	0.59	3479.33	4909	900.22	29.12	3526.57	0.11	0.17	4354.51	4426	900.88	1.62
11	20	140	80	3411.59	0.66	3540.66	4633	900.20	23.58	3516.09	5.86	0.17	4305.53	4385	900.89	1.81
12	20	160	80	3357.59	0.66	3521.45	4445	900.17	20.78	3442.63	6.78	0.42	4198.45	4389	901.72	4.34
13	20	140	60	3431.68	0.59	3567.30	4740	906.10	24.74	3566.01	0.08	0.07	4507.57	4668	900.38	3.44
14	20	160	60	3379.11	0.61	3497.41	4894	900.22	28.54	3500.70	0.09	0.11	4431.08	4471	900.48	0.89
15	20	140	80	3350.75	0.62	3417.21	4638	902.77	26.32	3457.59	4.73	0.26	4409.00	4409	629.40	0.00
16	20	160	80	3299.21	0.64	3475.06	4437	900.15	21.68	3406.97	5.46	0.29	4229.27	4402	901.28	3.92
17	20	140	60	2929.15	0.64	3049.45	4951	900.15	38.41	3096.93	68.87	0.13	4674.00	4674	160.79	0.00
18	20	160	60	2867.78	0.61	2941.52	4727	900.17	37.77	3038.31	80.98	0.24	4475.00	4475	227.71	0.00
19	20	140	50	2508.09	0.61	2629.81	4590	902.04	42.71	2652.43	21.02	0.16	4420.00	4420	103.99	0.00
20	20	160	50	2455.01	0.61	2574.96	4797	901.00	46.32	2578.59	24.46	0.28	4312.04	4414	900.29	2.31
21	20	140	50	3020.74	0.66	3245.80	5137	900.18	36.82	3241.31	6.43	0.07	4835.00	4835	38.94	0.00

(continued)

Table 1 (continued)

Inst.	n	W	Q	Model (1)–(12)						Model (13)–(19)						
				z_{RL}	t_{RL}	lb	ub	t_{UB}	gap	z_{RL}	t_g	t_{RL}	lb	ub	t_{UB}	gap
22	20	160	50	2949.96	0.62	3134.84	4969	901.71	36.91	3161.96	7.63	0.13	4675.55	4795	900.31	2.49
23	20	140	50	3058.84	0.58	3130.66	5146	901.45	39.16	3265.44	26.26	0.04	4744.00	4744	43.26	0.00
24	20	160	50	2984.49	0.59	3075.07	4882	900.18	37.01	3186.08	31.27	0.07	4656.00	4656	67.14	0.00
25	25	140	50	3778.16	1.05	3792.85	–	909.58	–	3903.70	10.87	0.24	4682.11	4751	901.15	1.45
26	25	160	50	3710.46	1.00	3722.81	6712	900.26	44.54	3833.08	12.49	0.38	4554.87	4646	900.81	1.96
27	25	140	50	3378.47	1.03	3389.69	5506	909.45	38.44	3615.80	16.32	0.17	4764.00	4764	116.82	0.00
28	25	160	50	3305.20	0.97	3334.35	–	913.82	–	3507.23	19.12	0.57	4598.88	4764	901.47	3.47
29	25	140	60	4283.93	1.01	4322.40	7384	900.43	41.46	4456.56	1.35	0.15	4817.00	4817	46.43	0.00
30	25	160	60	4213.33	1.01	4276.19	6876	907.18	37.81	4376.88	1.56	0.45	4683.61	4803	901.29	2.49
31	25	140	60	4150.87	1.03	4178.40	5409	900.26	22.75	4297.64	0.79	0.19	4669.18	4767	901.04	2.05
32	25	160	60	4085.55	1.03	4092.00	7110	900.19	42.45	4214.16	0.91	0.33	4527.01	4767	901.20	5.03
33	25	140	60	3654.99	1.01	3691.40	7585	905.24	51.33	3877.60	384.26	0.25	4779.00	4779	560.22	0.00
34	25	160	60	3577.31	1.00	3615.25	7420	905.71	51.28	3773.96	441.71	0.64	4643.14	4774	900.23	2.74
35	25	140	50	3159.18	1.01	3166.00	6910	900.24	54.18	3354.05	344.90	0.33	4670.22	4755	900.15	1.78
36	25	160	50	3092.86	0.99	3131.67	6633	913.87	52.79	3253.57	396.97	0.76	3503.94	4777	901.90	26.65
37	25	140	50	3766.16	0.98	3780.75	–	900.20	–	4036.95	70.72	0.17	5083.00	5083	269.30	0.00
38	25	160	50	3682.22	0.97	3709.32	–	904.97	–	3929.26	82.49	0.20	4864.00	4864	139.04	0.00
39	25	140	50	3422.25	1.01	3528.78	7501	900.25	52.96	3623.33	244.34	0.29	4889.00	4889	439.84	0.00
40	25	160	50	3355.86	1.03	3527.48	6688	901.62	47.26	3516.24	282.99	0.59	3898.86	4970	901.52	21.55
1	20	140	50	4102.90	0.75	4243.68	4821	900.19	11.98	4220.60	1.10	0.14	4431.00	4431	21.48	0.00
2	20	160	50	4048.67	0.72	4180.86	4684	900.17	10.74	4177.21	1.17	0.25	4320.23	4430	900.50	2.48

3	20	140	70	4041.16	0.64	4134.01	4676	900.17	11.59	4161.58	50.60	0.35	4327.69	4384	900.91	1.28
4	20	160	70	3990.65	0.59	4067.83	4621	900.15	11.97	4126.90	58.23	0.63	4187.73	4374	901.76	4.26
5	20	140	50	4254.64	0.62	4310.68	4788	902.31	9.97	4452.00	2.20	0.07	4681.00	4681	400.36	0.00
6	20	160	50	4189.73	0.59	4246.68	4479	900.25	5.19	4347.02	2.56	0.16	4472.00	4472	22.15	0.00
7	20	140	70	4177.13	0.61	4210.62	4753	900.18	11.41	4346.06	155.02	0.24	4424.00	4424	159.18	0.00
8	20	160	70	4115.92	0.66	4150.27	4714	900.18	11.96	4269.82	180.26	0.38	4372.46	4424	894.55	1.17
9	20	140	60	4091.69	0.58	4193.51	4716	900.22	11.08	4236.77	0.10	0.11	4431.00	4431	8.28	0.00
10	20	160	60	4038.11	0.61	4125.63	4718	900.22	12.56	4178.71	0.11	0.19	4364.48	4426	900.57	1.39
11	20	140	80	4042.85	0.64	4137.37	4418	900.14	6.35	4163.41	5.82	0.22	4276.15	4385	900.15	2.48
12	20	160	80	3992.55	0.69	4072.23	4405	900.25	7.55	4134.48	6.76	0.33	4196.36	4387	900.09	4.35
13	20	140	60	4080.59	0.59	4228.23	5054	905.02	16.34	4231.15	0.08	0.09	4537.00	4537	55.04	0.00
14	20	160	60	4029.93	0.59	4176.39	4711	903.01	11.35	4188.32	0.09	0.16	4440.73	4471	900.47	0.68
15	20	140	80	4014.69	0.69	4105.45	4633	903.46	11.39	4152.15	4.63	0.20	4330.24	4409	900.12	1.79
16	20	160	80	3967.37	0.64	4056.60	4503	905.08	9.91	4138.81	5.40	0.36	4234.47	4402	900.89	3.81
17	20	140	60	4160.48	0.64	4255.33	5062	900.18	15.94	4333.73	68.61	0.10	4674.00	4674	135.92	0.00
18	20	160	60	4103.24	0.80	4183.77	5091	900.18	17.82	4284.75	80.26	0.24	4475.00	4475	87.88	0.00
19	20	140	50	4080.34	0.72	4203.90	4719	908.30	10.92	4226.93	21.07	0.18	4420.00	4420	32.62	0.00
20	20	160	50	4027.86	0.66	4125.87	5028	900.22	17.94	4177.64	24.48	0.37	4347.47	4414	901.06	1.51
21	20	140	50	4328.32	0.62	4424.90	5050	907.35	12.38	4568.72	6.47	0.08	4835.00	4835	32.54	0.00
22	20	160	50	4257.48	0.64	4432.56	4897	900.17	9.48	4485.41	7.63	0.17	4670.36	4795	900.65	2.60
23	20	140	50	4327.32	0.62	4389.74	5073	900.20	13.47	4514.76	25.94	0.05	4744.00	4744	27.60	0.00
24	20	160	50	4254.49	0.64	4373.42	4656	901.26	6.07	4439.88	30.69	0.09	4656.00	4656	48.13	0.00

(continued)

Table 1 (continued)

				Model (1)–(12)						Model (13)–(19)						
Inst.	*n*	*W*	*Q*	z_{RL}	t_{RL}	*lb*	*ub*	t_{UB}	*gap*	z_{RL}	t_g	t_{RL}	*lb*	*ub*	t_{UB}	*gap*
25	25	140	50	4269.12	1.11	4275.27	–	902.31	–	4398.97	10.73	0.27	4675.25	4751	900.65	1.59
26	25	160	50	4201.42	1.09	4203.27	–	900.20	–	4329.62	12.35	0.49	4552.01	4646	900.12	2.02
27	25	140	50	4357.11	0.98	4447.17	5516	901.73	19.38	4580.81	16.27	0.20	4764.00	4764	89.45	0.00
28	25	160	50	4285.83	1.03	4369.81	–	903.68	–	4478.98	18.92	0.80	4599.30	4759	901.32	3.36
29	25	140	60	4347.52	1.00	4407.50	7265	900.28	39.33	4514.70	1.35	0.17	4817.00	4817	458.31	0.00
30	25	160	60	4276.64	0.97	4360.26	–	900.22	–	4435.38	1.56	0.32	4671.01	4800	901.01	2.69
31	25	140	60	4271.49	1.03	4328.35	–	901.35	–	4420.88	0.79	0.21	4654.18	4767	900.61	2.37
32	25	160	60	4206.29	1.01	4230.27	–	912.18	–	4351.79	0.90	0.39	4526.00	4767	901.09	5.06
33	25	140	60	4374.35	1.06	4403.43	5767	900.28	23.64	4581.74	382.16	0.29	4779.00	4779	387.52	0.00
34	25	160	60	4300.78	1.12	4377.00	7535	900.25	41.91	4493.35	443.39	1.16	4703.91	4774	901.54	1.47
35	25	140	50	4263.53	1.01	4278.66	9817	900.20	56.42	4458.98	355.09	0.36	4655.75	4755	900.06	2.09
36	25	160	50	4198.13	1.05	4229.79	–	900.22	–	4373.88	403.45	0.81	4557.76	4606	900.11	1.05
37	25	140	50	4492.47	1.014	4667.72	–	900.23	–	4754.01	71.59	0.12	5083.00	5083	338.08	0.00
38	25	160	50	4408.97	1.05	4463.85	–	909.89	–	4646.35	83.54	0.26	4864.00	4864	149.90	0.00
39	25	140	50	4346.90	1.01	4587.28	–	900.22	–	4562.43	247.52	0.36	4889.00	4889	302.53	0.00
40	25	160	50	4280.69	1.00	4385.18	5451	902.48	19.55	4476.18	285.20	0.66	4711.41	4880	900.25	3.45

Table 2 Computational results (Part II)

					Model (13)–(19) without cuts		Model (13)–(19) with cuts	
Inst.	*n*	*W*	*Q*	*best ub*	ub_h	t_{ub_h}	ub_h	t_{ub_h}
1	20	140	50	4431	4679	8.85	4435	5.62
2	20	160	50	4430	4750	10.90	4436	11.13
3	20	140	70	4384	4437	60.85	4599	52.90
4	20	160	70	4374	4430	72.75	4492	61.59
5	20	140	50	4681	4753	4.08	4745	7.19
6	20	160	50	4472	4785	10.32	4722	4.11
7	20	140	70	4424	4732	162.40	4437	159.39
8	20	160	70	4424	4766	195.47	4652	201.34
9	20	140	60	4431	4694	8.59	4692	0.63
10	20	160	60	4426	4692	30.60	4439	13.86
11	20	140	80	4385	4431	6.58	4644	19.74
12	20	160	80	4387	4407	37.57	4445	14.38
13	20	140	60	4537	4726	2.98	4714	31.49
14	20	160	60	4471	4741	4.93	4483	6.84
15	20	140	80	4409	4457	10.16	4719	5.46
16	20	160	80	4402	4412	35.88	4419	6.86
17	20	140	60	4674	4684	72.67	4684	70.88
18	20	160	60	4475	4475	111.28	4787	83.98
19	20	140	50	4420	4434	51.58	4421	22.34
20	20	160	50	4414	4448	55.13	4663	25.95
21	20	140	50	4835	4835	17.94	5020	8.04
22	20	160	50	4795	4823	38.42	4807	8.58
23	20	140	50	4744	5048	27.87	5044	26.69
24	20	160	50	4656	4809	34.21	4744	31.63
25	25	140	50	4751	4807	12.73	5022	12.65
26	25	160	50	4646	4796	20.77	4791	14.68
27	25	140	50	4764	5023	47.17	4797	17.32
28	25	160	50	4759	4815	50.53	4783	21.39
29	25	140	60	4817	5131	6.09	5131	7.11
30	25	160	60	4800	5115	32.25	5091	32.20
31	25	140	60	4767	5063	31.88	5069	31.47
32	25	160	60	4767	4922	11.37	5032	31.82
33	25	140	60	4779	5382	388.27	5028	418.68
34	25	160	60	4774	4966	474.46	4842	452.50
35	25	140	50	4755	4792	381.08	4796	351.51
36	25	160	50	4606	4796	437.96	4782	438.14
37	25	140	50	5083	5409	81.02	5329	80.45
38	25	160	50	4864	5067	114.17	4952	106.13
39	25	140	50	4889	5157	279.87	5085	248.76
40	25	160	50	4880	5172	318.70	4904	292.49

(19) provides the optimal solution within the time limit, and for the other cases, it gives solutions with very small optimality gaps. The results given in Table 2 further illustrate the fact that model (13)–(19) can be used to generate efficiently good incumbents for the problem.

Acknowledgements This work was supported by FEDER funding through the Programa Operacional Factores de Competitividade—COMPETE and by national funding through the Portuguese Science and Technology Foundation (FCT) in the scope of the project PTDC/EGE-GES/116676/2010 (Ref. COMPETE: FCOMP-01-0124-FEDER-020430).

References

1. Akca, Z., Berger, R., Ralphs, T.: Modeling and solving location routing and scheduling problems. Technical Report (2008)
2. Clautiaux, F., Alves, C., Valério de Carvalho, J.M.: A survey of dual-feasible and superadditive functions. Ann. Oper. Res. **179**, 317–342 (2010)
3. Lin, C., Chow, C., Chen, C.: A location-routing-loading problem for bill delivery services. Comput. Ind. Eng. **43**, 5–25 (2002)
4. Lin, C., Kwok, C.R.: Multi-objective metaheuristics for a location-routing problem with multiple use of vehicles on real data and simulated data. Eur. J. Oper. Res. **175**, 1833–1849 (2006)
5. Macedo, R.: Models and algorithms for hard optimization problems. Ph.D. thesis, Universidade do Minho, Braga (2011)
6. Macedo, R., Alves, C., Valério de Carvalho, J.M.: An exact approach for the location routing problem with multiple routes. In: Proceedings of the International Conference on Engineering UBI2011 "Innovation and Development" - ICEUBI2011, Covilhã (2011)

An Agent-Based Approach to Schedule Crane Operations in Rail-Rail Transshipment Terminals

Sam Heshmati, Zafeiris Kokkinogenis, Rosaldo J.F. Rossetti, Maria Antónia Carravilla, and José Fernando Oliveira

1 Introduction

A rail-rail transshipment terminal (RRTT) consists of a number of rail tracks, where trains are positioned in bundles to be served, and gantry cranes move containers between different trains, without exchanging the wagons. This is a complex system which requires proper schedule and collaboration of resources.

It is important to deal with uncertainty (e.g. unscheduled shipping request) in operational planning and control of RRTTs. The use of simulation can support a proper design of the system while dealing with uncertainty. Nevertheless, the complexity of this type of terminals makes it hard to model the problem using standard simulation approaches. An alternative that in recent years has received a lot of attention is the agent-based modeling and simulation (ABMS) approach. Although ABMS has been applied in logistic problems, literature does not report significant work with RRTT problems.

This study aims at analyzing transshipment processes in RRTT, from an operational point of view. This involves decisions on the position of containers on outbound trains; on the assignment of container moves to cranes, as predefined overlapping areas for the cranes are considered; and on sequence of the transshipments. The goal is to minimize the total transshipment time. We consider the ABMS approach to model the container transshipment problem in RRTTs. The use of the

S. Heshmati (✉) • M.A. Carravilla • J.F. Oliveira
INESC TEC, Faculty of Engineering, University of Porto, Porto, Portugal
e-mail: s.heshmati@fe.up.pt; mac@fe.up.pt; jfo@fe.up.pt

Z. Kokkinogenis • R.J.F. Rossetti
Artificial Intelligence and Computer Science Laboratory, Faculty of Engineering, University of Porto, Porto, Portugal
e-mail: pro08017@fe.up.pt; rossetti@fe.up.pt

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_12

agent metaphor allows us to come up with solutions in highly dynamic scenarios, while fixed dispatching rules do not have the same capability. The objective is to assign positions for all containers on outbound trains or on the yard, and determine the sequence of container movements for each crane.

The RRTT is described as an essential part of a hub-and-spoke architecture and an emerging technology in railway systems [9]. Several studies in the literature claim that the global rail-rail transshipment problem is too complicated to be monolithically solved [6, 10]. We can decompose the problems of RRTT into seven sub-problems: (1) Schedule the service slots of trains. (2) Assign a destination to each train. (3) Assign each train to a parking position. (4) Determine container positions on outbound trains (load plan). (5) Assign container moves to cranes. (6) Schedule the shuttle cars in the sorter. (7) Determine the sequence of container moves per crane.

The first decision problem of the RRTT was studied by Boysen et al. [5, 7], with the objective of minimizing the split moves, the number of revisits by trains, and the number of delayed containers. Problem (5) is studied by Boysen and Fliedner [4] with the objective of minimizing the makespan using a dynamic programming procedure. Alicke [1] tackled sub-problems (5), (6), and (7), as a constraint satisfaction problem. In [10], Souffriau et al. jointly assign the destination to trains (2), determines the load plan (4) and the sequence of the transshipments (7).

In the literature, there are some works using the agent-based metaphor to address issues in the seaport terminal operations and in inter-modal terminals, at a strategic planning level and policies evaluation [2, 3, 8].

However questions regarding the integration of the above problems into a holistic procedure still remained unaddressed. To the best of our knowledge, ABMS has not been applied to tackle any of the aforementioned problems in RRTTs. In this study we are aiming at the integrated resolution of three sub-problems: (4) Determine container positions on outbound trains, (5) Assign container moves to cranes, and (7) Determine the sequence of container moves per crane.

The remainder of this paper is structured as follows: in the next section the overview of the problem and the simulation model are presented. Section 3 discusses preliminary results of the model. Finally, the conclusion and opportunities for future work are explained in Sect. 4.

2 Problem Description and Simulation Model

This section describes a typical transshipment yard and the three constitutive sub-problems of Scheduling Crane Operations (SCO) problem. A transshipment yard consists of a number of parallel tracks, gantry cranes, and a quay with ground vehicles to move the containers on the quay. The trains (one per track) arrive in bundles to the yard, to be served simultaneously, and the destination of each train has already been assigned. A train is composed of a number of wagons, which carry containers. The gantry cranes process the container moves in parallel.

The SCO problem has to be solved for every bundle of trains arriving at the yard and the objective is to minimise the makespan. It includes three sub-problems of a typical RRTT. First, the SCO problem assigns each container to a position on the outbound train with the proper destination in such a way that the overall containers' movement distance is minimized. Next, it assigns container moves to cranes. Finally, the schedule of container moves per crane is determined.

Position Containers on Outbound Trains The objective of this sub-problem is to determine the load pattern in such a way that the total container movement is minimized. A train is composed by a number of wagons of different types and capacity. Associated to the wagon type is length and configuration. The wagon configuration is a set of position slots with a specific length. Containers are defined in terms of destination, type, and length. A container can be placed on a position slot if the position slot is free and its length is equal to the container's length.

A transshipment starts from the position slot where the container is currently located on the inbound train and ends at the position slot of the outbound train to which the container will have to be moved. Containers' destination/outbound train are predefined. This sub-problem finds the final position slot on the outbound train.

Assign Container Moves to Cranes The objective is to split the overall workload evenly among the cranes. *Direct transshipment* is a type of transshipment when the initial and the final position slots of a transshipment are within the working area of one crane. Otherwise, the first crane has to deliver the container to the ground vehicle on the quay. The ground vehicle carries the container along the yard to the working area of the second crane where the transshipment can be concluded. This type of operation is called *indirect transshipment*. It is better to have direct transshipments than indirect ones because of the cost of extra pick-ups and drops.

The cranes can operate with disjoint or overlapping areas. For the present work we have considered the latter. The transshipment assignment should be done dynamically depending on the utilization of the cranes.Using overlapping working areas, the number of indirect transshipments will be reduced, but also it has to be ensured that the cranes do not collide. This is done by blocking the overlapping area when a crane enters it. Large overlapping areas may lead to high idle times of the crane which is not operating in the overlapping area.

Sequence Container Moves per Crane Finally, it is necessary to determine the sequence of container moves per crane. The objective of this sub-problem is to minimize the total transshipment time and, at the same time, to avoid collisions between two neighbouring cranes. Due to the empty travelling between two transshipments, crane moves are asymmetric in distance. A transshipment job can be carried out only when the destination position is free. In this study we are trying to avoid potential deadlocks (which happens when the crane needs to swap the place of two containers) by determining the final position of a transshipment only when the move is going to be executed.

The model implements a service slot of a real RRTT. The yard has eight tracks, two cranes with an overlapping influence area, a ground vehicle, and a quay. Each position slot can hold one container with the same length of the slot. The working area of each crane agent is 60 % of the length of the longest train, results in a 20 % overlap of crane areas. The traveling time is replaced by the Chebyshev distance since it is assumed that the lateral and longitudinal speeds of the cranes are equal.

To setup the simulated environment, the model generates the terminal, quay, trains, position slots, containers, and cranes based on input file. The input file includes the following data for each entity: **Trains:** track number, length, and number of wagons in each train. **Position slots:** train number, X coordinate, and type. **Containers:** coordinates, type, and destination. The cranes and ground vehicles are implemented as agents. The cranes can move over the trains on the yard both in the X and Y directions, and the ground vehicles can move on the quay along the X axis. To prevent the collision between cranes, an overlapping area is defined that takes the state *blocked* whenever a crane enters it.

The model implements two rules for agent cranes to decide which transshipments to choose to process. Each time an agent crane starts a transshipment job, randomly applies one of these rules. The **First rule** is to pick up the nearest unprocessed container, and the **Second rule** is to pick up the nearest unprocessed container on the train with the highest number of unprocessed containers. Whenever a crane cannot find a free position slot on the destination train, inside its own influence area, it will drop the container on the ground vehicle on the quay. The ground vehicle moves the container to the area of the other crane, which will process the container. If still there is no free position slot, the container will remain on the yard, waiting for the next train with the same destination. The simulation terminates when all the existing trains in the yard are served and ready to leave the yard.

The present model is implemented in Netlogo [11], an agent-based simulation platform and programming language. It is discrete in time and space as the agents are situated in a discretized world (i.e. grid) and perform actions in discrete time steps (i.e. ticks). A tick (the simulation time step) is assumed to correspond to 1 s of real-world time while the space is divided in squares of 1 m side length. Differently from discrete event simulation (DES) approach which is typically used in the logistic domain, in the agent-based model the monitoring and control are distributed among the agents and they build their local decision making processes. Another key difference is that DES is built around networks of priority queues of (time, event) pairs, while in the present model there is no concept of queues.

3 Simulation Results

The simulation model has been validated on a set of problem instances taken from the literature [10]. For this terminal, instances have 8 tracks, 5 destinations and 5 trains, with 20 wagons each train. Two wagon types are considered: 60 and 85 ft. Possible container lengths are 20, 30 and 40 ft. The load factor is defined as the total length of the containers divided by the total length of the trains. The following range of load factors is used: $\{0.1, 0.2, \ldots, 0.9\}$ (step of 0.1). There are five instances for each load factor. The results presented below are averaged over these five instances per load factor. The computational experiments were run on a computer with an Intel i7-2620M, with a clock speed of 2.70 GHz and 8 GB of RAM.

Following Souffriau et al. [10], and in order to be possible to compare the two approaches, the results of the simulation model are presented as the improvement of the objective function (minimizing the total transshipment time) over a random solution. We implemented the approach of [10] to solve the sub-problem (2) and used the results as input to sub-problem (4). Table 1 presents the load factor, the number of containers (# Containers), the number of containers left on the quay (# Containers on yard), the execution time, service time of each train, and the average percentual improvement. As expected these results show that the problem becomes harder to solve when the load factor increases. Moreover, there is a considerable improvement (greater than 70 %) for instances with a load factor equal or smaller than 0.6, which then drops markedly until a value of 12 % for a load factor of 0.9. The average improvement of the agent-based simulation model is 63 %, which is similar to the result obtained by Souffriau et al. [10]: 62 % of improvement.

However the computational time of our approach is significantly lower, when compared to theirs. In Souffriau et al. [10] the Variable Neighbourhood Descent (VND) algorithm was run with a number of perturbations (of the solutions in local optima) ranging from 0 to 10, with increasing computing times. Even if we take the results with 0 perturbations, which only deliver an average improvement of 53.9 %, the execution times range from 0.4 s, for a load factor of 0.1–3278 s, for a load factor of 0.9 (Intel Xeon with a clock speed of 2.5 GHz and 4 GB RAM). Even taking into account the different hardware platforms, it is significant that for a load factor of 0.9 our approach is 40 times faster than the VND algorithm in its fastest configuration. If the more time-consuming configuration is taken, this factor rises up to a value of approximately 160.

An important performance indicator is also the number of containers left on the quay. In instances with a load factor greater than 0.6 there are indeed containers left behind on the quay, with a maximum value of 14.2 (in average) for instances with a load factor equal to 0.9. This behaviour is common to the approach of Souffrriau

Table 1 Simulation results averaged over five instances per load factor

Load factor	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
# Containers	29	55.6	83.4	110.6	138.2	168.2	198	237	251.4
# Containers on yard	0	0	0	0	0	0	0.8	6.4	14.20
Execution times (s)	0.60	0.90	1.34	1.89	2.39	3.13	9.54	32.96	77.70
1st train service time	1073	1788	2862	4334	5739	7439	13,352	26,150	36,633
2nd train service time	1155	1987	2992	4530	5938	7764	13,792	27,044	37,853
3rd train service time	1298	2255	3111	4791	6206	8053	14,373	27,796	39,138
4th train service time	1480	2371	3206	4923	6547	8616	15,127	28,105	41,466
Average cost	1708	2697	3367	5265	6982	9374	16,178	28,568	44,476
Average improvement (%)	72.47	77.51	76.99	78.14	75.52	72.91	58.7	39.82	12.00

et al. [10], although with lower values: 0.2, 2.8, and 4.0 containers left on the quay, for load factors 0.7, 0.8 and 0.9 respectively.

4 Conclusions

This paper deals with the scheduling crane operations (SCO) problem in rail-rail transshipment terminals (RRTT). In this combinatorial optimization problem the containers have to be transshipped among trains by multiple cranes. The SCO problem involves three decisions: (1) to define the position of containers on outbound trains, (2) to assign transshipments to cranes, and (3) to sequence transshipments for each crane. The problem is tackled by an agent-based simulation model, in which agent cranes deal with the three decisions and transship the containers. A set of problem instances taken from literature is used to validate the model and the results are compared against [10]. The model achieved up to 75 % improvement, a value similar to [10] results, but in a fraction of the time used by the authors.

Future work includes applying dynamic yard assignment, where no specific area is defined for cranes and agent cranes cooperate and coordinate through negotiation processes to avoid collision. The results are expected to minimize the main drawback of the current model that is the number of containers left on the quay.

Acknowledgements Project "NORTE-07-0124-FEDER-000057" is funded by the North Portugal Regional Operational Programme , under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF), and by national funds, through the Portuguese funding agency, Fundação para a Ciência e a Tecnologia (FCT). Zafeiris Kokkinogenis is funded by FCT under the PhD grant SFRH/BD/67202/2009.

References

1. Alicke, K.: Modeling and optimization of the intermodal terminal mega hub. OR Spectr. **24**, 1–18 (2002)
2. Bergkvist, M., Davidsson, P., Persson, J.A., Ramstedt, L.: A hybrid micro-simulator for determining the effects of governmental control policies on transport chains. In: Multi-Agent and Multi-Agent-Based Simulation, pp. 236–247. Springer, Berlin/Heidelberg (2005)
3. Böcker, J., Lind, J., Zirkler, B.: Using a multi-agent approach to optimise the train coupling and sharing system. Eur. J. Oper. Res. **131**, 242–252 (2001)
4. Boysen, N., Fliedner, M.: Determining crane areas in intermodal transshipment yards: the yard partition problem. Eur. J. Oper. Res. **204**, 336–342 (2010)
5. Boysen, N., Jaehn, F., Pesch, E.: Scheduling freight trains in rail-rail transshipment yards. Transp. Sci. **45**, 199–211 (2011)
6. Boysen, N., Fliedner, M., Jaehn, F., Pesch, E.: A survey on container processing in railway yards. Transp. Sci. **1655**, 1–18 (2012)

7. Boysen, N., Jaehn, F., Pesch, E.: New bounds and algorithms for the transshipment yard scheduling problem. J. Sched. **15**, 499–511 (2012)
8. Henesey, L., Davidsson, P., Persson, J.A.: Agent based simulation architecture for evaluating operational policies in transshipping containers. In: Multiagent System Technologies, pp. 73–85. Springer, Berlin/Heidelberg (2006)
9. Macharis, C., Bontekoning, Y.: Opportunities for or in intermodal freight transport research: a review. Eur. J. Oper. Res. **153**, 400–416 (2004)
10. Souffriau, W., Vansteenwegen, P., Berghe, G., Oudheusden, D.: Variable neighbourhood descent for planning crane operations in a train terminal. In: Sörensen, K., Sevaux, M., Habenicht, W., Geiger, M.J. (eds.) Metaheuristics in the Service Industry. Lecture Notes in Economics and Mathematical Systems, vol. 624, pp. 83–98. Springer, Berlin/Heidelberg (2009)
11. Wilensky, U.: NetLogo: Center for Connected Learning and Computer-Based Modeling. Northwestern University, Evanston (1999). http://ccl.northwestern.edu/netlogo/

Vertical and Horizontal Collaboration in Inventory and Transportation

Benedikt De Vos and Birger Raa

1 Introduction

Rising competition and fuel prices give shippers a hard time staying profitable and force them to look for more efficient ways to organize logistic activities [1]. Since a company's performance depends on both its own decisions and those of other supply chain players, a more coordinated approach to manage the supply chain is in order.

Vertical collaboration occurs between companies on subsequent supply chain levels [6]. Different forms exist, like *Collaborative Planning, Forecasting and Replenishment* (CPFR) or *Vendor Managed Inventory* (VMI). Most existing research focuses on this type of collaboration. Horizontal collaboration takes place between companies on the same supply chain level [6]. Distribution constitutes a large part of logistic costs and cooperation among shippers may entail major savings. However, research on this kind of cooperation or on joint horizontal and vertical cooperation, is limited and focuses on empirical evidence of its importance [1]. Cruijssen [1] has shown that the impediments for successful cooperation among shippers are persistent. They cannot find partners because of their reluctance to share information and because they do not know how to divide the gains fairly among the partners [1].

A fair gain sharing mechanism is crucial to assure sustainability. All partners should be better off within the coalition, than individually, or they will leave the coalition [5]. *Cooperative Game Theory* (CGT) is often used to model the negotiation process between the parties. Three solution concepts are encountered regularly in the existing literature, the core, the Shapley Value and the nucleolus

B. De Vos (✉) • B. Raa
Ghent University, Technologiepark 903, 9052 Zwijnaarde, Belgium
e-mail: Benedikt.DeVos@UGent.be; Birger.Raa@UGent.be

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_13

(e.g. [3, 4]). Research on CGT within horizontal distribution collaboration is limited though [1].

2 Problem Description

Although distribution is a key supply chain driver, cooperation has rarely been applied in this area [2]. This paper adds to the existing literature by introducing both horizontal and vertical collaboration in logistics in a specific problem setting. The goal is to raise cost-efficiency through cooperation in inventory management and distribution and to divide the gains using GCT to guarantee long-term viability.

The considered supply chain has three levels: retailers, shippers and *Logistics Service Providers* (LSPs). Each shipper has its own set of retailers. They can replenish them using their own vehicles, or shippers can cooperate through an LSP. The LSP then distributes the goods of multiple shippers to create savings through order volumes and geographical proximity. Volume savings are created by coordinating multiple shippers' orders at one stocking point, such that the fixed cost, which is incurred each time an order is made, can be shared by the shippers. Transport savings originate from serving retailers in the same area. We assume that LSPs proactively look for shippers with a high synergy potential, which is referred to as *insinking* [2].

2.1 Calculations Cost Savings Shippers

To determine the savings, the cost of each shipper working individually is calculated first. The total annual cost comprises the annual inventory cost (holding and order cost) and the annual transport cost. Each shipper has an individual order cost F and holding cost H to determine his Economic Order Quantity (EOQ) and his annual number of orders n (cfr. (1)). The transport cost is found through the minimal-cost routes to his retailers. Transport costs comprise a distance cost per travelled kilometer, a service cost per retailer and a loading cost per vehicle.

$$EOQ = \sqrt{\frac{2 \cdot D \cdot F}{H}}, n = \frac{D}{EOQ} \tag{1}$$

$$TC_{Si} = n \cdot F + \frac{EOQ}{2} \cdot H + \text{Transportation cost} \tag{2}$$

We include an example to clarify our methodology. It holds two LSPs, each serving three shippers, which in their turn have three retailers, each one with its own daily demand D. We assume that all shippers sell similar products.

Shippers can enter a coalition S and cooperate through an LSP, such that no information has to be shared with competitors. When cooperating, shippers no longer incur a holding cost and their order cost changes. The LSP charges them a fixed fee per order FF and a variable fee per transported unit VF. An LSP can serve a coalition with one individual shipper. The shipper's holding cost becomes the sum of his customer-specific order cost CSF, which is incurred when outsourcing to an LSP, and the fixed fee FF (cfr. (3)). This results in a new number of orders and a new EOQ. His total cost equals the sum of the new inventory cost and the variable fee (cfr. (4)). When the coalition contains only one shipper, the savings v(S) represent the savings obtained by outsourcing distribution to an LSP (cfr. (5) and Table 1).

$$F_{co} = FF_{LSPi} + CSF_{Si} \tag{3}$$

$$TC_{SCo} = n \cdot F_{co} + \text{Variable Fee} \tag{4}$$

$$v(S) = MAX\{TC_{SCo} - TC_{Si}; 0\} \tag{5}$$

When an LSP serves multiple shippers, the savings v(S) originate from both outsourcing and horizontal cooperation among the shippers. The coalition of shippers incurs a new order cost F_{co} (cfr. (6)), including the customer-specific order cost of all shippers in the coalitions and the fixed fee. This results in a new number of orders, new EOQ, new total cost (cfr. (7)) and new savings (cfr. (8)). Table 2 shows the results of two possible coalitions for both LSPs in our example.

$$F_{co} = FF_{LSPi} + \sum_{i \in S} CSF_{Si} \tag{6}$$

$$TC_{co} = n \cdot F_{co} + \text{Variable Fee} \tag{7}$$

$$v(S) = \text{MAX}\{TC_{co} - \sum_{i \in S} TC_{Si}; 0\} \tag{8}$$

Table 1 Cost shippers: individual costs and costs in coalition with one shipper

Ship	D	F_{ind}	EOQ	Ship cost	Co	F_{co}	EOQ	LSP cost	Total fee	Ship savings
1	10	233.25	1409	34,622.63	{1}	216	1350	32,701.60	34,114.50	508.13
2	10	206.37	1328	35,703.25	{2}	217	1350	32,696.38	34,115.75	1587.50
3	10	224.04	1281	33,487.04	{3}	215.5	1200	25,107.20	34,381.50	0.00
4	10	206.65	1265	37,666.13	{4}	215	1200	28,307.70	34,407.00	3259.13
5	10	251.54	1328	33,127.30	{5}	214	1200	29,398.50	34,530.00	0.00
6	10	247.00	1281	31,346.96	{6}	211.8	1350	27,384.40	33,869.65	0.00

Table 2 Coalition cost shippers: coalitions with multiple shipper

Co	F_{co}	EOQ1	EOQ2	EOQ3	LSP cost	Total fee	Supply chain savings	LSP profit	Ship savings
{2,3}	247.5	0	982	982	54,612.24	63,390.36	14,578.04	8778.12	2421.75
{1,2,3}	278.5	831	831	381	84,959.83	94,446.56	18,853.09	9486.73	4818.55
{5,6}	243.8	0	982	982	51,074.62	65,626.72	13,399.66	12,391.70	1007.94
{4,5,6}	276.8	831	831	831	71,895.74	96,838.41	30,244.64	21,701.77	8542.88

Table 3 Coalition multiple LSPs: best coalitions LSP1 and LSP2

Coalition	Total cost	Total fee	LSP profit	Supply chain savings	Ship savings
{1,2,3,4,5,6}	139,474.62	179,192.00	39,717.38	66,478.68	18,304.62

2.2 *Logistic Service Provider Calculations*

Similar to shippers, LSPs have an individual order cost F and holding cost H. Their total cost also includes inventory and transport costs, which are calculated for each possible coalition. LSPs has to store the shippers' EOQs, which do not necessarily equal the LSPs' EOQ. The transport cost is determined by their minimal-cost routes. We assume that LSPs distribute the goods from their own depots. If a shipper both produces and distributes the goods, the finished products need to be transported to the LSP warehouse first. Hence, the LSP incurs an additional transport cost. The LSP profit is calculated by subtracting his costs from the shippers' fees (cfr. (10)). The LSPs' cost and profit for four coalitions can be found in Table 2.

$$TC_{LSP} = n \cdot F_{LSPi} + \sum_{i \in S} \frac{EOQ_i}{2} \cdot H + \text{Transportation cost} \tag{9}$$

$$P_{LSP} = \sum_{i \in S} \text{Variable Fee}_{Si} + \text{Fixed Fee}_{Si} - TC_{LSP} \tag{10}$$

It is possible for LSPs to collaborate too, but only if the shippers allow them to collaborate, which requires significant savings for the shippers. If allowed, the weighted transport fees of all LSPs (fixed and variable) are used to calculate the new EOQ and number of orders. The transport cost is found by the minimal-cost routes from multiple LSP depots. Results from our example are shown in Table 3.

The number of coalitions out of S shippers is $(2^S - 1)$. Since every coalition's cost has to be calculated, each additional shipper creates an exponential number of calculations. We use a heuristic method to get a good solution for the coalition, without performing all calculations. We identify the best coalitions per LSP and consider the shippers in these coalitions to form coalitions where all LSPs collaborate.

2.3 Coalition Selection

We aim to select the best coalition, which is the one minimizing the supply chain cost (cfr. Table 2: coalition {1,2,3} and {4,5,6}). The LSP tries to form the coalition with the highest profit, which requires appropriate fees, such that both the LSP and shippers benefit from the coalitions. In our example, the coalitions with the lowest supply chain cost also create the highest profit for the LSPs (cfr. Table 2).

It is important that the chosen coalitions do not increase the costs of the coalition members. If a shipper's cost increases, the next best coalition should be examined. In Coalition{1,2,3} for example, all three shippers should obtain a fair amount of the savings and their cost cannot be higher than when they work individually.

2.4 Savings and Profit Allocation

Once the savings are known, they are allocated among the shippers. We use the Shapley value, which is the marginal expected contribution of an additional player i in a coalition (cfr. (11)). It is important to take the marginal contribution of each shipper to the total gain into account, such that the allocation is perceived as fair. Other advantages are its computational simplicity and its unique savings allocation, which makes it practical in real-life situations. Table 4 shows the savings allocation according to the Shapley value for our example.

$$SV_i(N, v) = \sum_{S \subset N; i \notin S} \frac{(|S|!|N| - |S| - 1)!}{|N|!} [v(S \cup \{i\} - v(S)] \tag{11}$$

If several LSPs collaborate, profit has to be fairly divided between them too. However, the Shapley value cannot be used to do so. We want to know the additional value created by an LSP, while the additional coalition players are shippers. Hence, a ratio based on the Shapley values of the shippers is proposed to divide the LSPs' profit. Each LSP's profit is calculated by multiplying the total profit with his ratio.

$$\text{ratio}_{LSP_j} = \frac{\sum_{i \in S, LSP_j} (SV_i)}{\sum_{i \in S} (SV_i)} \tag{12}$$

Table 4 Shapley values shippers coalitions {1,2,3} and {4,5,6}

Savings {1,2,3}	Shipper	Shapley value	Savings {4,5,6}	Shipper	Shapley value
4818.55	1	1553.79	8542.88	4	6044.56
	2	2630.74		5	1557.07
	3	634.02		6	941.25

This way, the LSPs' profit depends on the shippers' savings, which incites the LSPs to pursue high overall supply chain savings. If the savings and profit would not be linked, the LSPs would be less inclined to pursue a high overall supply chain performance. Now, the LSP that brings in the largest savings for the supply chain earns the largest profit share.

3 Conclusion

We studied the gains of joint horizontal and vertical cooperation among shippers and LSPs in inventory and distribution. Synergies were created through order volumes and joint routing. To coordinate the supply chain, we applied game theory and used the Shapley value as a distribution key for the shippers' savings. This value resulted in a more justified solution than rules of thumb do. The profit allocation among LSPs cannot be done by the Shapley value. Instead, we used a ratio that takes into account the contribution of the LSP's shippers to the total savings. This ratio incites the LSP to pursue better global supply chain performance, rather than high individual profits.

Preliminary results on larger datasets indicate that similar results can be expected for more realistic settings. However, they also indicate that several parameters are potentially crucial for the coalition formation and savings creation and allocation. Especially the location of the retailers and LSPs and the LSP fees have a pronounced impact on the savings and coalition formation and on the savings allocation. The shippers' costs and retailers' demand have no clear impact on the coalitions, but they do influence the profit distribution, since higher individual shipper costs creates higher opportunities to create savings and high demand rates creates higher profits. More in-depth research into larger datasets is necessary. Furthermore, other problem variants like the pick-up and delivery problem should be investigated to look for additional cost savings. More research possibilities lie in sensitivity and robustness of the coalitions and savings.

References

1. Cruijssen, F., Cools, M., Dullaert, W.: Horizontal cooperation in logistics: opportunities and impediments. Transp. Res. E Logist. Transp. Rev. **43**(2), 129–142 (2007)
2. Cruijssen, F., Borm, P., Fleuren, H., Hamers, H.: Supplier-initiated outsourcing: a methodology to exploit synergy in transportation. Eur. J. Oper. Res. **207**(2), 763–774 (2010)
3. Frisk, M., Gothe-Lundgren, M., Jornsten, K., Ronnqvist, M.: Cost allocation in collaborative transportation. Eur. J. Oper. Res. **205**(2), 448–458 (2010)
4. Lozano, S., et al.: Cooperative game theory approach to allocating benefits of horizontal cooperation. Eur. J. Oper. Res. **229**(2), 444–452 (2013)
5. Ozener, O., Ergun, O.: Allocating costs in a collaborative transportation procurement network. Transp. Sci. **42**(2), 146–165 (2008)
6. Simatupang, T.M., Sridharan, R.: The collaborative supply chain. Int. J. Logist. Manag. **13**(1), 15–30 (2002)

Part III
Production

Experimental Study and Statistical Analysis of Human Factors' Impact in Cell Production System

Yanwen Dong and Xiying Hao

1 Introduction

The Cell Production (CP) or Cellular Manufacturing (CM) has become an integral part of lean manufacturing systems; many organizations have applied CP concepts in manufacturing and service processes. Implementation of CP has been shown to achieve significant improvements in product quality, scheduling, space utilization, control of operations and employee morale [8].

The numerous techniques and methods have focused on technical aspects of CP, such as the best groupings for products, parts or machine clusters, selecting tools, determining process flow. Although technical problems of CP have been thoroughly researched and many mathematical and computer based approaches have been reported, there is a singular absence of articles that deal with the human factors in CP because human related issues are typically difficult to quantify [1, 2].

It has been found that for successful implementation of CP, people who will eventually operate, manage, support and maintain the manufacturing cells should actively participate in their design and development [9]. Wemmerlov et al. [10] surveyed 46 user plants with 126 cells and concluded that substantial benefits could be achieved from CP but that implementation is not simply a rearrangement of the factory layout; it is a complex reorganization that involves organizational and human aspects. They emphasized that most of the problems faced by companies implementing cells were related to people, not technical issues.

This study aims to investigate the impact of human factors in CP and contribute to the literature on CP from the following two viewpoints:

Y. Dong (✉) • X. Hao
Cluster of Science and Technology, Fukushima University, Fukushima City 960-1296, Japan
e-mail: dong@sss.fukushima-u.ac.jp; haoxiying201314@yahoo.co.jp

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_14

1. As most of previous researches applied questionnaire survey or case study methods, it is only possible to evaluate human factors' impact comparatively and empirically. Different from previous researches, this paper will apply the experimental study method to evaluate the impact of human factors more precisely. We conduct a CP experiment, measure operation time, and then assess statistically the effect of human factors based on the time measurement results.
2. With the exception of our previous researches [3, 5, 7], no studies have been published in which the impact of workers' aptitude was quantitatively examined. In this paper, we design a questionnaire to measure the operators' aptitude, and then conduct the regression analysis to clarify the relationship between operators' aptitude and productivity of manufacturing cells.

This paper is organized as follows. At first, we introduce the experiment and questionnaire design. Then we conduct a factor analysis to specify personal aptitude based on the answers of the questionnaire. Next, we investigate the impact of personal aptitude on the productivity of CP. At last, we show some concluding remarks.

2 Experiment and Questionnaire Design

2.1 Cell Production Experiment

We designed a laboratory experiment to examine the impact of human factors on the performance of CP. We use a toy robot that built up of LEGO Mindstorms as the virtual good. It consists of 106 pieces of parts and the assembling process is divided into 17 tasks. The experiment is carried out along with the following steps:

Step 1. Instruction: Giving the operators some assembly manuals, the instructor demonstrates the assembling tasks of the toy robot through assembling it practically in front of the operators. Following the instructor's demonstration, the operators learn the sequence and techniques to assemble the toy robot, and assemble one toy robot by oneself.

Step 2. Assembling and measurement: After the instruction, the operators assemble the toy robot in the mode of one-person cell. When doing the assembling tasks, the operation time was measured.

Step 3. Repeat: The assembly time to assemble a toy robot is calculated as the sum of operation times of all tasks. In order to investigate the learning effect, the assembling operation and time measurement are repeated five times.

Table 1 Questionnaire

Execution	No	Question item
Before the assembling operation	Q_1	I'm interested in this experiment
	Q_2	I feel this experiment that's fun
	Q_3	This experiment seems difficult
	Q_4	I think this experiment is significant for me
	Q_5	This experiment is quite troublesome
	Q_6	I like experimental subjects including this experiment
	Q_7	I like making something
	Q_8	I am going to actively participate in this experiment
	Q_9	I like the classroom lectures better than an experiment
	Q_{10}	Fine work is my favorite
	Q_{11}	I think my hand is deft
After the assembling operation	Q_{12}	I worked actively during this experiment
	Q_{13}	I was not good at fine work
	Q_{14}	This experiment was interesting
	Q_{15}	This experiment was meaningful
	Q_{16}	The assembly tasks were repeated too many times
	Q_{17}	This experiment was more enjoyable than others
	Q_{18}	This experiment was difficult
	Q_{19}	The instructor's instruction could be understood well
	Q_{20}	In order to raise efficiency, I had devised for myself
	Q_{21}	The experiment manual was very comprehensible
	Q_{22}	This experiment was monotonous and boring
	Q_{23}	I felt my hand is deft

2.2 *Questionnaire Design*

According to Hackman and Oldham's Job Characteristics Model (JCM) [4] and our observations from the experiment [6], we designed a questionnaire to investigate personnel-related aspects, such as deftness, participation, attitude, etc. of the operators. As showed in Table 1, the questionnaire sheet consists of 23 questions and the operators are required to answer it before and after the assembling operation. When answering questionnaire, a five-point Likert scale was employed.

3 Factor Analysis and Personal Aptitude

There are 68 operators attended the experiment, who are the students taking the experimental lesson entitled as Industrial System Laboratory in Fukushima University. Sixty-one of them returned both valid answers to the questionnaire and valid time measurement. To investigate the impact of operators' personal aptitude on production performance, we conducted an exploratory factor analysis as follows.

We have n ($n = 61$) operators and m ($m = 23$) questions in the questionnaire. Let q_{ij} be the answer of question Q_j ($j = 1, 2, \ldots, m$) from operator i ($i = 1, 2, \ldots, n$). Suppose that the latent structure underlying the answers of the questionnaire includes p factors and f_{ik} denotes the factor score of operator i for the factor k ($k = 1, 2, \ldots, p$), the corresponding unique factors are u_{ij} ($i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, m$), the observed variables q_{ij} can be expressed as linear functions of these p factors:

$$q_{ij} = \sum_{k=1}^{p} f_{ik} a_{kj} + u_{ij} \tag{1}$$

Where a_{kj} is the factor loading of the question Q_j on the factor k. Using the principal axis factoring method and varimax rotation with Kaiser Normalization, we could extract five factors and obtain the factor loadings, as shown in Eq. (2).

$$A = (a_{jk}) = \begin{pmatrix}
Q_1 & -0.132 & \mathbf{0.690} & 0.202 & 0.041 & -0.145 \\
Q_2 & 0.105 & \mathbf{0.737} & 0.255 & 0.143 & -0.201 \\
Q_3 & \mathbf{-0.632} & -0.007 & 0.169 & -0.258 & 0.088 \\
Q_4 & -0.118 & \mathbf{0.531} & 0.258 & 0.046 & -0.050 \\
Q_5 & -0.118 & -0.231 & 0.030 & 0.089 & \mathbf{0.758} \\
Q_6 & 0.118 & 0.186 & 0.106 & \mathbf{0.851} & -0.078 \\
Q_7 & 0.351 & \mathbf{0.366} & 0.263 & 0.329 & -0.220 \\
Q_8 & 0.075 & \mathbf{0.687} & 0.304 & 0.289 & -0.158 \\
Q_9 & -0.195 & -0.157 & 0.126 & \mathbf{-0.639} & -0.019 \\
Q_{10} & \mathbf{0.882} & 0.015 & 0.299 & 0.053 & -0.020 \\
Q_{11} & \mathbf{0.899} & -0.015 & 0.195 & 0.066 & -0.007 \\
Q_{12} & 0.207 & \mathbf{0.804} & 0.129 & 0.095 & -0.015 \\
Q_{13} & \mathbf{-0.855} & -0.028 & -0.090 & -0.119 & -0.008 \\
Q_{14} & 0.256 & 0.248 & \mathbf{0.765} & 0.130 & -0.183 \\
Q_{15} & 0.061 & 0.310 & \mathbf{0.747} & 0.049 & -0.052 \\
Q_{16} & 0.071 & 0.054 & -0.224 & -0.306 & \mathbf{0.590} \\
Q_{17} & 0.202 & 0.355 & \mathbf{0.662} & -0.076 & -0.091 \\
Q_{18} & \mathbf{-0.404} & -0.304 & 0.192 & -0.203 & 0.276 \\
Q_{19} & 0.009 & 0.493 & \mathbf{0.522} & 0.118 & -0.129 \\
Q_{20} & 0.335 & \mathbf{0.543} & 0.217 & 0.075 & 0.096 \\
Q_{21} & 0.096 & 0.407 & \mathbf{0.591} & -0.239 & -0.029 \\
Q_{22} & 0.010 & -0.143 & -0.388 & 0.039 & \mathbf{0.483} \\
Q_{23} & \mathbf{0.837} & 0.133 & 0.110 & -0.043 & 0.031
\end{pmatrix} \tag{2}$$

According to Eq. (2) and Table 1, the following five factors ($p = 5$) can be extracted.

- Deftness (factor 1): it represents the operator's deftness to finish a fine and difficult work.
- Positiveness (factor 2): it describes if the operators are interested and can participate actively in this experiment.
- Meaningfulness (factor 3): it represents the task significance or how meaningful the operators felt the tasks.
- Preferrer (factor 4): it describes if the operators prefer manual works.
- Passiveness (factor 5): it represents if the operators were passive to do a fine and monotonous work.

After the factor extraction, the factor score f_{ik} of operator i ($i = 1, 2, \ldots, n$) for the factor k ($k = 1, 2, \ldots, p$) can also be calculated. These factor scores will be used in the next section as the evaluation values of the operators for the five factors.

4 Productivity and Personal Aptitude

To investigate how the five factors extracted above affect the performance of CP, we represent the assembly time of the operator i ($i = 1, 2, \ldots, n$) at the r-th ($r = 1, 2, \ldots, 5$) experience as t_{ri}, and then conduct five kind of regression analysis, as shown in Eq. (3). Here, the dependent variable is t_{ri}, and the independent variables are the factor scores f_{ik} ($k = 1, 2, \ldots, p$).

$$t_{ri} = \sum_{k=1}^{p} b_{rk} f_{ik} + e_{ri} \tag{3}$$

Where b_{rk} is the regression coefficient, and e_{ri} is the error term ($i = 1, 2, \ldots, n$; $r = 1, 2, \ldots, 5; k = 1, 2, \ldots, p$).

Suppose f_{ik} and t_{ri} have been standardized, then we can derive the standardized regression coefficients according to the least squares method. As the result, the following five standardized regression equations could be obtained.

$$t_{1i} = -\mathbf{0.323} f_{i1} + 0.007 f_{i2} + 0.008 f_{i3} - \mathbf{0.356} f_{i4} + 0.039 f_{i5} \tag{4}$$

$$t_{2i} = -\mathbf{0.288} f_{i1} - 0.037 f_{i2} - 0.051 f_{i3} - \mathbf{0.263} f_{i4} + 0.087 f_{i5} \tag{5}$$

$$t_{3i} = -\mathbf{0.203} f_{i1} - 0.127 f_{i2} - 0.085 f_{i3} - \mathbf{0.288} f_{i4} + 0.068 f_{i5} \tag{6}$$

$$t_{4i} = -\mathbf{0.395} f_{i1} + 0.016 f_{i2} - 0.063 f_{i3} - \mathbf{0.419} f_{i4} + 0.093 f_{i5} \tag{7}$$

$$t_{5i} = -\mathbf{0.344} f_{i1} - 0.094 f_{i2} + 0.080 f_{i3} - \mathbf{0.284} f_{i4} + 0.031 f_{i5} \tag{8}$$

Table 2 The significance of the standardized regression coefficients

Independent variable	Dependent variable				
	t_{1i}	t_{2i}	t_{3i}	t_{4i}	t_{5i}
f_{i1}	0.008	0.023	0.106	0.001	0.005
f_{i2}	0.951	0.763	0.311	0.886	0.432
f_{i3}	0.943	0.683	0.494	0.563	0.503
f_{i4}	0.004	0.037	0.023	0.000	0.020
f_{i5}	0.745	0.482	0.584	0.393	0.797

We show also the significance level (P value) each regression coefficient in Table 2. According to the results of Table 2, it is clear that the scores of factor 1 and factor 4 correlated significantly to the assembly times. That is, the preferrer to manual works (factor 4) and the deftness for a fine and difficult work (factor 1) have a significant impact on the productivity of CP. Meanwhile, as the scores of factor 2, factor 3 and factor 5 didn't correlated significantly to the assembly times, we could not verify statistically the impact of the task significance (meaningfulness) and operators' attitude (positiveness and passiveness) on performance of CP.

5 Concluding Remarks

This paper has designed a laboratory experiment of CP to measure the productivity of manufacturing cells. Meanwhile, a questionnaire was executed to measure the operators' aptitude. Based on the answers of the questionnaire, a factor analysis was conducted and five human factors: deftness, positiveness, meaningfulness, preferrer and passiveness, were extracted. However, among the five factors, only two of them have been verified statistically that they have significant impact on the productivity of CP. From this result, it was clarified that the productivity of cell production depends significantly on two aptitudes of the operators: the preferrer to manual works and the deftness for a fine and difficult work. On the other hand, the task significance (meaningfulness) and operators' attitude (positiveness and passiveness) have not significant relation to the productivity of CP.

Because only two-fifths of the factors have been verified statistically that they have significant relation to the productivity of CP, the questionnaire should be improved so that it can extract as many significant factors as possible. Furthermore, we are going to apply other methods such as structural equation modeling to find more effective latent structure underlying the questionnaire.

Acknowledgements This work was partly supported by Grant-in-Aid for Scientific Research (C) from the Japan Society for the Promotion of Science under Grant No. 25380497.

References

1. Bopaya, B., Poonsiri, A., Kim, L.N., Bryan, A.N., Wipawee, T.: Human related issues in manufacturing cell design implementation, and operation: a review and survey. Comput. Ind. Eng. **48**, 507–523 (2005)
2. Charlene, A.Y., Harold, J.S.: Cellular manufacturing for small businesses: key cultural factors that impact the conversion process. J. Oper. Manag. **20**, 593–617 (2002)
3. Dong, Y., Haraguchi, H., Hao, X.: Structural equation modelling of human factors and their impact on productivity of cellular manufacturing. Innov. Supply Chain Manag. **8**, 1–7 (2014)
4. Fried, Y., Ferris, G.R.: The validity of the job characteristics model: a review and meta-analysis. Pers. Psychol. **40**, 287–322 (1987)
5. Hao, X., Haraguchi, H., Dong, Y.: An experimental study of human factors' impact in cellular manufacturing and production line system. Information. **16**, 4509–4526 (2013)
6. Haraguchi, H., Dong, Y.: An experimental study of human factors' impact in cellular manufacturing and production line system. In: Proceedings of Asian Conference of Management Science and Applications (ACMSA2011), Sanya, p. 381 (2011)
7. Haraguchi, H., Inoue, K., Dong, Y.: Structural equation modeling of human factors and their impact on productivity of cellular manufacturing. In: the 21th International Conference on Production Research (ICPR21), Stuttgart, 31 July–4 August 2011
8. Liu, C.G., Lian, J., Yin, Y., Li, W.: Seru seisan- an innovation of the production management mode in Japan. Asian J. Technol. Innov. **18**, 89–113 (2010)
9. Olorunniwo, F.O., Udo, G.J.: Cell design practices in us manufacturing firms. Prod. Invent. Manag. J. **37**, 27–33 (1996)
10. Wemmerlov, U., Johnson, D.J.: Cellular manufacturing at 46 user plants: implementation experiences and performance improvements. Int. J. Prod. Res. **35**, 29–49 (1997)

Planning Production and Workforce in a Discrete-Time Financial Model: Optimizing Cash-Flows Released

Pedro Martins

1 Introduction

Production planning is a central theme of discussion within Management Science. It also concentrates the attention of the Operations Research community, namely on lot-sizing problems, to which an extensive number of scientific works and a large amount of real-world successful applications are available in the literature (see, e.g., [1, 4, 7, 8]). The integration of cash-flows and workforce along the production process has also been discussed, namely in [2, 3, 5, 6].

In the present paper, we propose a mixed integer linear programming formulation that also attempts to handle these three planning processes in a single framework, acting together in a discrete-time stream. In our approach, the production process is not as comprising as the versions discussed in the former papers, as it lies on a single-item and single-level basis. However, our model combines the three mentioned processes with strategies for cash-flows released, namely dividends, while satisfying a given sustainability condition or a final outcome condition. The objective is to maximize the entire amount of cash-flows released outwards. In addition, we do not force the sales to entirely meet the demand, but using the demand level as an upper limit for the sales strategy. The mentioned simplification on the production process is only to simplify the discussion, in order to emphasize the trade-off relationship among the three processes. The entire system can be enlarged with the various features usually considered in lot-sizing planning problems.

P. Martins (✉)
Polytechnic Institute of Coimbra, ISCAC, Quinta Agrícola - Bencanta, 3040-316 Coimbra, Portugal

Operations Research Center (CIO), University of Lisbon, Lisbon, Portugal
e-mail: pmartins@iscac.pt

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_15

In the next section, we describe the integrated financial/workforce/production problem under discussion, to which a formulation is proposed in Sect. 3. A case oriented study is discussed in Sect. 4. The paper ends with a section on conclusions.

2 Financial/Workforce/Production Planning Problem

Given a stream of discrete time periods, ranging from period 1 to n, we want to describe production, workforce and cash-flows in each of these periods, along the entire time horizon. The production process is single-item and single-level, that is, it involves a single product on a single processing unit. However, the entire system can be seen on a two-level scheme. The first level involves a financial sequence, while the second one runs on a production stream. Capital to borrow and workforce are additional resources for sustaining the two processes.

Considering the production stream, we know the demand in each period, which is not required to be fully attained, acting just as an upper limit for the sales, while assuming that the unsatisfied demand is lost. We also know, for each period, the unitary net profit of the sales, the unitary cost of keeping the product in stock, the fixed cost of production and the capacity of production.

Workforce is required for covering the production. Thus, we have a sequence of time intervals defining overlapping shifts, where each shift lasts m periods. The first work-shift starts in period 1 and ends in period m, then the second shift starts in period 2 and ends in period $m+1$, and so on. The last shift starts in period $n-m+1$ so that no worker is on duty after period n. Workers contracts have a work-shift duration, that is, m periods. We want to determine the number of workers to hire for each shift such that the production is covered. In this case, we know the cost of each worker in each period and the production rate of a worker, that is, the number of units that a worker can produce in each period.

The cash-flows are expected to interact with the previously described processes. They should pay production and workforce costs, while being fed by the sales' profits. We can assume an initial cash income supplied by the shareholders. In addition, the company can borrow a loan in every period of the planning stream, each one with an h period's maturity. Amortizations are paid in equal amounts along the h period's interval. We assume that both contracts and amortizations are made at the beginning of the period, and that all loans are entirely paid at the end of the planning horizon. Thus, the last period for borrowing is period $n-h$. There is a given interest rate for the loan starting in each period. We also consider an upper limit for the entire debt in each period. Cash-flows released outwards are also determined along the planning stream. The problem also comprises a sustainability condition or a final outcome condition, forcing the cash-balance at the end of the planning horizon (period n) to meet a given lower limit cash amount. We set this lower limit to be, at least, the initial capital supplied by the shareholders at the beginning plus profits.

The objective is to maximize the sum of the cash-flows released outwards, during the entire planning horizon.

3 Mathematical Formulation

In order to model the problem we start defining the parameters, the variables and then introduce the mixed-integer linear programming formulation. We use two letters for representing the parameters (apart from index ranges) and a single letter for the variables. As mentioned above, n denotes the planning horizon, m represents the shifts duration and h denotes the loan's maturities. We assume that $n > \max\{m, h\}$.

Parameters:

$dm_t \equiv$ demand in period t, $t = 1, \ldots, n$
$ps_t \equiv$ unitary net profit of the sales (deducting fixed costs and salaries) in period t, $t = 1, \ldots, n$
$cs_t \equiv$ unitary cost of the stock in period t, $t = 1, \ldots, n$
$fc_t \equiv$ production fixed costs in period t, $t = 1, \ldots, n$
$cp_t \equiv$ capacity of production in period t, $t = 1, \ldots, n$
$cw_t \equiv$ cost of a worker (per period) arrived in the shift started in period t, $t = 1, \ldots, n - m + 1$
$pr \equiv$ labor rate (number of units that a worker can produce (in any period))
$ic_0 \equiv$ initial capital supplied by the shareholders before starting the process
$ir_t \equiv$ interest rate of the loan started in period t, $t = 1, \ldots, n - h$
$dl \equiv$ upper limit for the entire debt in each period
$sp \equiv$ profits expected in the sustainability/outcome condition, as a proportion of parameter ic_0

Variables:

$p_t \equiv$ production in period t, $t = 1, \ldots, n$
$s_t \equiv$ stock at the end of period t, $t = 1, \ldots, n$ ($s_0 = 0$, by assumption)
$y_t = \begin{cases} 1, & \text{if } p_t > 0 \\ 0, & \text{otherwise} \end{cases}, \quad t = 1, \ldots, n$
$w_t \equiv$ number of workers in the shift starting in period t, $t = 1, \ldots, n - m + 1$
$v_t \equiv$ cash balance at the end of period t, $t = 1, \ldots, n$
$b_t \equiv$ capital borrowed in period t, $t = 1, \ldots, n - h$
$r_t \equiv$ cash-flows released outwards (e.g., dividends) in period t, $t = 1, \ldots, n$

Formulation:

$$\text{maximize} \quad \sum_{t=1}^{n} r_t \tag{1}$$

$$\text{subject to} \quad 0 \leq s_{t-1} + p_t - s_t \leq dm_t \ , \quad t = 1, \ldots, n \tag{2}$$

$$p_t \leq cp_t \cdot y_t \ , \quad t = 1, \ldots, n \tag{3}$$

$$p_t \leq \sum_{j=\max\{1,t-m+1\}}^{\min\{n-m+1,t\}} pr \cdot w_j \quad , \quad t = 1, \ldots, n \tag{4}$$

$$\sum_{j=\max\{1,t-h+1\}}^{t} \frac{j-t+h}{h} \cdot b_j \leq dl \quad , \quad t = 1, \ldots, n-h \tag{5}$$

$$ic_0 + ps_1 \cdot (p_1 - s_1) + b_1 = cs_1 \cdot s_1 + fc_1 \cdot y_1 + cw_1 \cdot w_1 + r_1 + v_1 \tag{6}$$

$$v_{t-1} + ps_t \cdot (s_{t-1} + p_t - s_t) + b_t =$$

$$= cs_t \cdot s_t + fc_t \cdot y_t + \sum_{j=\max\{1,t-m+1\}}^{\min\{n-m+1,t\}} \left(cw_j \cdot w_j\right) +$$

$$+ \sum_{j=\max\{1,t-h\}}^{\min\{t-1,n-h\}} \left(\frac{1+ir_j \cdot (j-t+h+1)}{h} \cdot b_j\right) + r_t + v_t \quad , \quad t = 2, \ldots, n \tag{7}$$

$$v_n \geq (1 + sp) \cdot ic_0 \tag{8}$$

$$p_t, s_t, v_t, r_t \geq 0 \quad , \quad t = 1, \ldots, n \quad ; \quad b_t \geq 0 \quad , \quad t = 1, \ldots, n-h \tag{9}$$

$$y_t \in \{0, 1\} \quad , \quad t = 1, \ldots, n \quad ; \quad w_t \in \mathbb{N}_0 \quad , \quad t = 1, \ldots, n-m+1 \tag{10}$$

The variable v_t should be ignored in the equalities (7) for $t = n-h+1, \ldots, n$. We have left them in the model in order to simplify the exposition. The set of constraints (2) model the production stream, where the amount sold in period t ($s_{t-1} + p_t - s_t$) is bounded by the demand (dm_t). Inequalities (3) impose a limit on the production in each period, whenever production is on, which will activate the associated fixed cost. Also, constraints (4) relate production to workforce availability in each period. These constraints are also bounding the production in each period. Then, inequalities (5) impose an upper limit (dl) on the sum of the debt in each period. Further, constraints (6) and (7) describe cash-flow conservation, where (6) involves period $t = 1$ and (7) characterize the remaining periods. In these equalities, we set all the cash income in the left-hand-side and the outgoing cash in the right-hand-side. The first summation in the right-hand-side of equalities (7) represents the total amount of salaries to pay in period t, while the second summation represents the amortizations and the interests also to be paid in period t. The last constraint state the sustainability/outcome condition, setting a lower limit on the cash balance at the end of the planning horizon, guaranteeing that at the end, the process will return all the capital invested plus profits.

4 Discussing an Application

In this section we propose a fictitious example, in order to simulate some aspects of the three processes (production/workforce/cash-flows) interaction. Each time period represents a month and the planning horizon includes 50 periods, thus, $n = 50$. We also consider that each shift has a 5 months duration ($m = 5$) and that the loans last 10 months (maturities $h = 10$). In addition, we assume that there is no significant inflationary effect during the entire time horizon.

Following the usual life time stream of a product, we consider the four stages for the demand: introduction, growth, maturity and decline. In the present example, we assume that the first period demand is equal to 1000 units ($dm_1 = 1000$). Then, the demand grows at a 3 % rate during the first 5 months (introduction term), it passes to a 7 % rate during the next 15 months (growth term), and slows down during the next 20 months with a growing rate of 1 % (maturity term). Then, it declines to a −5 % rate (decline term). In addition, we assume that the product will be off-line at the end of the planning horizon, which suggests that the last condition (constraint (8)) should be seen as a lower limit for a final cash outcome.

The unitary net profits (in euros) of the sales (ps_t) are randomly generated, following a Normal distribution with $\mu = 10$ and $\sigma = 2$. The same way, the interest rates (in percentage) of the loans (ir_t) are also randomly generated following a Normal distribution with $\mu = 0.5$ and $\sigma = 0.002$. All the remaining parameters were assumed to be constant along the planning horizon, taking the following values.

$$cs_t = 1 \text{ euro}, \ t = 1, \ldots, n \qquad pr = 100 \text{ units}$$

$$fc_t = 1000 \text{ euros}, \ t = 1, \ldots, n \qquad ic_0 = 20{,}000 \text{ euros}$$

$$cp_t = 5000 \text{ units}, \ t = 1, \ldots, n \qquad dl = 50{,}000 \text{ euros}$$

$$cw_t = 850 \text{ euros}, \ t = 1, \ldots, n - m + 1 \qquad sp = 80\,\%$$

Considering these data, and using ILOG/CPLEX 11.2 for solving the model, the optimum solution value is equal to 140,576.86 euros. This is the maximum capital that can be released outwards (for dividends, for instance) during the entire planning horizon, such that the required conditions are met, namely, the final outcome goal that forces the last period cash balance to bring the initial capital supplied by the shareholders (ic_0) plus 80 % of profits over the mentioned capital. Thus, besides the capital released outwards, the company will take to the future (variable v_n) a cash balance equal to 36,000 euros. In addition, the workforce strategy suggests hiring in

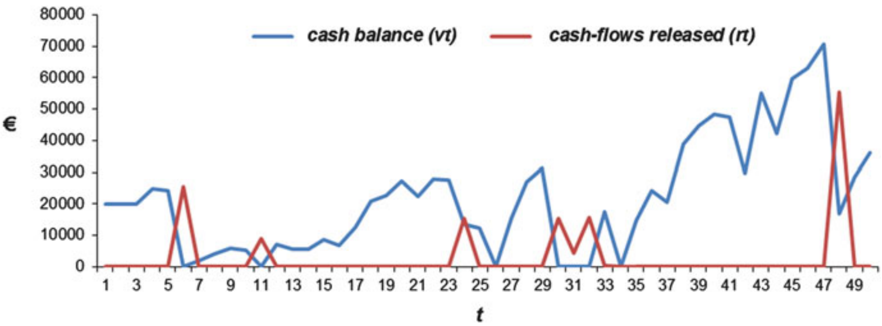


Fig. 1 Cash balance and cash-flows released outwards along the entire planning stream

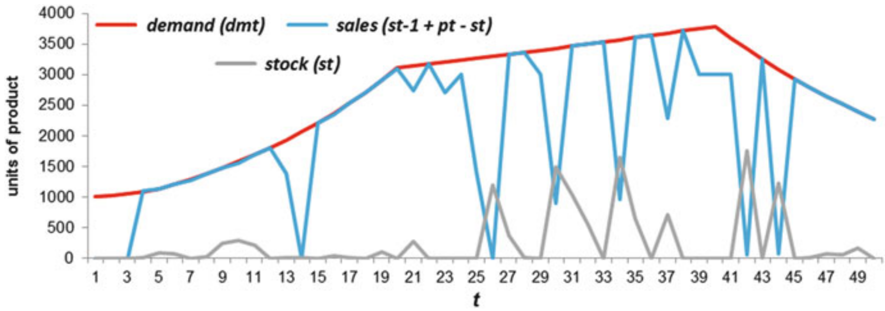


Fig. 2 Demand, sales and stocks along the entire planning stream

the shifts starting in periods 4, 5, 8, 9, 15–21, 24, 25, 27–29, 31–34, 36–39, 41–43, 45 and 46. In these shifts, we should hire 11, 1, 2, 14, 22, 2, 1, 2, 3, 22, 2, 6, 6, 13, 5, 6, 6, 13, 5, 2, 10, 13, 5, 2, 10, 1, 2, 4 and 21 workers, respectively. Also, the solution recommends borrowing no capital during the entire planning horizon.

Figure 1 represents the cash balance and the cash-flows released outwards along the entire planning stream. Figure 2 compares the demand, the effective sales and the stocks along the same stream.

5 Conclusions

The main motivation of the present paper is to bring an additional mathematical programming based tool for planning production, workforce and some relevant aspects involving cash-flows, exploring their interaction in a single framework. Using a small fictitious example, we have made a few steps pursuing the discussion of the entire system. Naturally, the model can be extended to larger dimensional and more complex problems, including additional features in the various streams involved.

Acknowledgements We thank the reviewers for the valuable comments and suggestions. This work was partially supported by FCT (project PEst-OE/MAT/UI0152).

References

1. Absi, N., Detienne, B., Dauzère-Pérès, S.: Heuristics for the multi-item capacitated lot-sizing problem with lost sales. Comput. Oper. Res. **40**(1), 264–272 (2012)
2. Fisk, J.: An interactive game for production and financial planning. Comput. Oper. Res. **7**(3), 157–168 (1980)
3. Kirka, Ö., Köksalan, M.: An integrated production and financial planning model and an application. IIE Trans. **28**, 677–686 (1996)

4. Loparic, M., Pochet, Y., Wolsey, L.A.: The uncapacitated lot-sizing problem with sales and safety stocks. Math. Program. **89**, 487–504 (2001)
5. Martins, P., Quelhas, A.P.: Workforce planning and financing on a production/capital discrete-time model. International Transactions in Operational Research (2015), in press (DOI: 10.1111/itor.12181)
6. Muñoz, M.A., Ruiz-Usano, R., Framiñán, E.A.: A mathematical programming model for the integration of aggregate production planning with short-term financial planning. In: Proceedings of the First World Conference on Operations Management (POM 2000), Sevilla (2000)
7. Pastor, R., Altimiras, J., Mateo, M.: Planning production using mathematical programming: the case of a woodturning company. Comput. Oper. Res. **36**(7), 2173–2178 (2009)
8. Pochet, Y.: Mathematical programming models and formulations for deterministic production planning problems. In: Jünger, M., Naddef, D. (eds.) Computational Combinatorial Optimization. Lecture Notes in Computer Science, vol. 2241, pp. 57–111. Springer, Berlin/Heidelberg (2001)

Evaluating Supply Chain Resilience Under Different Types of Disruption

Sónia R. Cardoso, Ana Paula Barbosa-Póvoa, Susana Relvas, and Augusto Q. Novais

1 Introduction

Currently SCs are more exposed to disruptions, due to their increasing complexity of operating worldwide. Such disruptions can be of two kinds, either "known-unknown" or "unknown-unknown". The former are associated with known occurrences, well documented by both academics and industry, and have a reasonably predictable probability of occurrence. Their impact is very low when compared to the second type of occurrences [10]. In order to deal with disruptions the need of incorporating resilience in supply chains and quantifying the effect of these events in their operations has been recognized by several companies and by academics [5]. A number of papers have been published in this area. However there is still a lack of quantitative models to address such occurrences [11]. SC resilience can be defined as the ability of a SC to return to its original state or move to a new more desirable state, after being disturbed [6]. Some authors have proposed strategies in order to build resilient supply chains. Fiksel [7] identifies four major characteristics: diversity, efficiency, adaptability and cohesion that should be pursued within SCs. Rice and Caniato [9] studied the necessity of investing in flexibility and redundancy. The majority of the literature published in the area is based on qualitative insights [2], and few authors have been trying to quantify resilience. Kima et al. [8] presented metrics derived from social network analysis and studied how these metrics affect

S.R. Cardoso • A.P. Barbosa-Póvoa • S. Relvas (✉)
CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, Av. Rovisco Pais, 1049-001 Lisboa, Portugal
e-mail: sonia.cardoso@tecnico.ulisboa.pt; apovoa@tecnico.ulisboa.pt; susanaicr@tecnico.ulisboa.pt

A.Q. Novais
LNEG, Laboratório Nacional de Energia e Geologia, Lisboa, Portugal

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_16

the reliability of supply chains. Adenso-Diaz et al. [1] applied thirteen metrics to SCs and concluded that the most relevant are node complexity, density and node criticality that affect negatively the supply chain reliability, and flow complexity that affects it positively. Carvalho et al. [4] used the lead time ratio and the total cost to compare the behaviour towards a disruption of a SC using two different strategies: one based on flexibility and another on redundancy. From the literature it can be concluded that, of quantitative work published in the area of the SC resilience, the majority addresses only one single type of disruption affecting one specific echelon of the network. Also, when resilience strategies were implemented no performance measures were applied. Thus, there are still a vast number of questions to investigate in this area.

2 Problem Definition

This work proposes a multi-product, multi-period MILP model for the design and planning of supply chains with uncertainty in products' demand, which will overcome some of the limitations found in the literature. It is based on a previous work of the authors [3] and is applied to two SCs, one a traditional forward network (case A) and the other a CLSC (case B), see Fig. 1. For further details on the model used as the readers are referred to the work of Cardoso et al. [3].

The suppliers send the raw materials to the plants where they are processed using a set of production processes. Then, the resulting intermediate products are sent to warehouses where the final orders are assembled and delivered to customers. Transshipment within plants and within warehouses is allowed. Case B is more complex than case A since it integrates reverse flows and has the possibility of sending products directly from plants to markets without passing through the warehouses. The reverse flow is originated in the markets and comprises three types of products: (1) "non-conforming" that are sent to warehouses to be repacked before being reintroduced in the SC, (2) "end-of-life" that are collected in period t, having been sold in an earlier period. These products are sent to the disassembling centers (located in the plants) to be disassembled or remanufactured and (3)

Fig. 1 Supply chain structures of cases A and B

"non-recoverable" that are sent directly to disposal. New entity installation or capacity expansions are allowed. Four different disruptions are studied: (D1) the two most important suppliers do not have raw material available during time period 2; (D2) 100 % decrease in the production capacity of the most important plant in time period 2; (D3) the 3PL (third party logistics) hired to operate those transportation links, with the highest quantity of products, between plants and warehouses goes out of business in time period 2; and (D4) the most important warehouse closes down in time period 2. Also and simultaneously with the disruption occurrence, demand uncertainty is considered using a scenario tree approach [12]. On the disruptions occurrence two options are studied: either they occur with certainty or with uncertainty. For the latter two more scenarios are considered corresponding to whether or not the disruption will effectively occur. To compare the resilience of both SCs, seven indicators are implemented, three are operational: expected net present value (ENPV), customer service level (CSL) and investment, and four are related to the network design: node and flow complexity, density and node criticality, which are determined by Eqs. (1)–(4).

$$Nodes_t = \sum_v X_{vt}, \forall t \tag{1}$$

$$FC_t = \sum_v \sum_w (Y_{vwt} + YNC_{vwt} + YEL_{vwt}), \forall t \tag{2}$$

$$Density_t = \frac{FC_t}{pt}, \forall t \tag{3}$$

$$NC_{vt} = \sum_{w:(w,v)\in F} Y_{vwt} + \sum_{w:(v,w)\in R} (YNC_{vwt} + YEL_{vwt}), \forall v, t \tag{4}$$

Equation (1) determines the node complexity ($Nodes_t$) through the sum of all binary variables (X_{vt}) that assume either the value 1 if entity v at time t belongs to the network or 0 otherwise. In Eq. (2) the flow complexity (FC_t) is calculated as the sum of the total forward (Y_{vwt}) and reverse flows (YNC_{vwt} for the non-conforming and YEL_{vwt} for end of life products). The density is determined in Eq. (3) as the ratio between the total number of flows that are established in a given time period (FC_t) and the total number of potential flows that could be established (pt). Node criticality (NC_{vt}) is the total number of critical nodes of a network. A node is considered critical when the sum of its inbound and outbound flows is higher than 10, which is calculated in Eq. (4).

3 Case Study

The developed model was applied to a SC operating in Europe. It involves an existing plant in Bilbao, one warehouse in Salamanca and four raw material suppliers located respectively in Badajoz, Barcelona, Frankfurt and Prague. Ten markets spread around different European countries are supplied. At the Bilbao plant, 12 manufacturing and 6 disassembling processes exist with an initial capacity of 600 ton. The warehouse has six assembling lines that assemble the final orders and exhibits a storage capacity of 500 ton. A reconfiguration of this structure is under study and the objective is to maximize the ENPV of the investment. The company considers the hypothesis of expanding the existing processes besides installing a new plant in Hamburg and two new warehouses in Lyon and Munich. On the demand forecasts, it was used the approach proposed by Tsiakis et al. [12] where it is assumed that demand is known for the first time period, while for the second period uncertainty exists modeled trough three possible branches: an optimistic that implies an increase of 10 % over the first time period forecast, with a probability of 0.25, a most likely with an increase of 3 % and a probability of 0.50, and a pessimistic that involves a reduction of 2 % in the demand with a probability of 0.25. For the third and last time period, and for each branch of the previous period, another three possibilities are considered: an optimistic which implies an increase of 5 % in the demand over the value assumed for the previous period, a most likely with an increase of 2 %, and a pessimistic with a decrease of 2 %. When the disruptions' occurrence is considered uncertain, D1 is assumed to have a probability of occurrence of 40 %, D2 of 10 %, D3 of 60 % and D4 of 25 %. Since each period represents 5 years, the modelling of three time periods results in a time horizon of 15 years, which seems to be more than enough for a supply chain to recover from a disruptive event. The results for the operational indicators are shown in Fig. 2 and Table 1. When a probability is associated to the occurrence of the disruption it is represented by the letter ST.

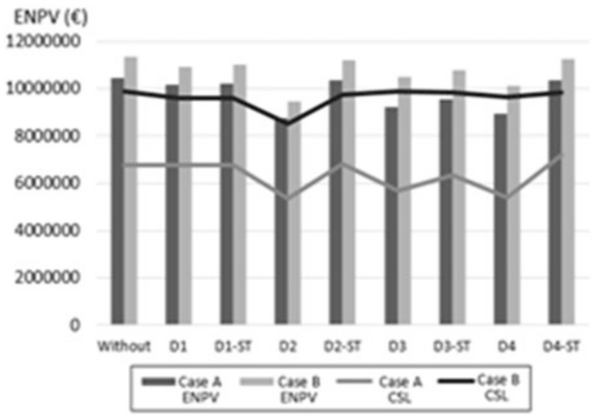


Fig. 2 ENPV and CSL for each case

Table 1 Relative deviations of ENPV and CSL under all disruptions (%)

	Case	D1	D1-ST	D2	D2-ST	D3	D3-ST	D4	D4-ST
ENPV	A	−2.80	−2.41	−16.37	−1.10	−12.02	−8.53	−14.72	−0.93
	B	−2.60	−2.25	−16.82	−1.12	−7.68	−5.23	−10.74	−0.83
CSL	A	−5.12	−2.66	−20.96	−2.21	−15.99	−6.50	−20.47	−0.20
	B	−2.67	−1.99	−13.75	−1.12	−0.03	−0.30	−2.03	−0.24

Table 2 Processes and storage capacities

	Production number	Processes capacity (un)	Disassembling number	Processes capacity (un)	Assembling number	Processes capacity (un)	Storage capacity (un)
Case A	22	1431.1	0	0	18	1462.4	509.4
Case B	24	1566.1	12	502.8	18	1800.9	578.4

When comparing both SCs without disruption (Fig. 2), it is found that case B achieves a higher ENPV, which is explained due to the fact that this network integrates the reverse flows, making use of the returned products and also presenting a higher flexibility in terms of transportation connections. Also a higher customer service level satisfying almost all demand is observed. When disruptions occur, case B presents always better results when compared to case A, presenting lower relative deviations between the under disruption and the no disruption scenarios (see Table 1). Disruptions 2 and 4 are the ones that have the higher impact in both networks, because these events are responsible by the destruction of one plant and one warehouse, respectively. When a probability is associated to the occurrence of the disruptions, the results are less affected. In terms of investment, case B always achieves higher investment values, as this network has to invest in extra capacity in order to meet almost 100 % of the demand. Both networks install the new plant and the two possible warehouses in time period 1, which are maintained during the entire planning horizon. The capacities and the number of the processes installed are higher in case B then in case A, as presented in Table 2.

Regarding the investment under disruptions, D2 implies the highest investment for both cases. This is because it is known beforehand that one plant closes down in time period 2. Hence, this must be compensated by assigning more capacity to the other plant in time period 1. Additionally, in time period 3, after the disruption occurs, an additional investment is made in a new plant. On the network design indicators Table 3 presents the values obtained for all tested scenarios.

The results for the network design indicators corroborate what was stated in the literature by Adenso-Diaz et al. [1]. Cases as case B that presents higher node, flow complexity and node criticality, and lower density in all scenarios, always exhibits the highest ENPV and CSL. Case B also shows lower relative deviations between cases with and without disruption, proving to be a more resilient supply chain by comparison with case A. Case B, since is more complex, has higher flexibility

Table 3 Network design indicators

	Case	Without	D1	D1-ST	D2	D2-ST	D3	D3-ST	D4	D4-ST
Node complexity	A	19	18	18	18	19	19	19	19	19
	B	21	21	21	21	21	21	22	21	22
Flow complexity	A	23	23	23	21	23	23	24	21	25
	B	65	67	64	66	63	68	71	65	65
Density	A	0.147	0.145	0.145	0.132	0.145	0.147	0.156	0.135	0.162
	B	0.129	0.133	0.127	0.131	0.124	0.133	0.140	0.128	0.129
Node criticality	A	0	0	0	0	0	0	0	0	0
	B	4	4	4	3	4	4	4	4	4

in terms of transportation links and, as such, has more choices of overcoming a disruption, if it occurs.

4 Conclusions

The main objective of this work was to develop a design and planning model that integrates demand uncertainty that may be applied to different network structure supply chains subject to disruptions that may affect different SC echelons. Through the usage of a set of indicators that comprise four network design indicators and three operational indicators, the studied networks were analysed in terms of their resilience. The results obtained for most complex SC suggest that the metrics used can be good indicators of resilience. For future work the authors aim to further support the presented conclusions by applying the developed model to other real-life supply chains. Furthermore the authors also aim to investigate the implementation of other types of disruption, with a view to develop a more in-depth analysis of the metrics studied. It is also intended to incorporate uncertainty at the level of the returns.

References

1. Adenso-Diaz, B., Mena, C., Garcia-Carbajal, S., Liechty, M.: The impact of supply-network characteristics on reliability. Supply Chain Manag. Int. J. **17**(3), 263–276 (2012)
2. Bode, C., Wagner, S., Petersen, K., Ellram, L.: Understanding responses to supply chain disruptions: insights from information processing and resource dependence perspectives. Acad. Manag. J. **54**(4), 833–856 (2011)
3. Cardoso, S., Barbosa-Póvoa, A., Relvas, S.: Design and planning of supply chains with integration of reverse logistics activities under demand uncertainty. Eur. J. Oper. Res. **226**(3), 436–451 (2013)
4. Carvalho, H., Barroso, A., Machado, V., Azevedo, S., Cruz-Machado, V.: Supply chain redesign for resilience using simulation. Comput. Ind. Eng. **62**, 329–341 (2012)

5. Chopra, S., Sodhi, M.: Managing risk to avoid supply-chain breakdown. MIT Sloan Manag. Rev. **46**, 52–61 (2004)
6. Christopher, M., Peck, H.: Building the resilient supply chain. Int. J. Logist. Manag. **15**(2), 1–13 (2004)
7. Fiksel, J.: Designing resilient sustainable systems. Environ. Sci. Technol. **37**(23), 5330–5339 (2003)
8. Kima, Y., Choi, T., Yan, T., Dooley, K.: Structural investigation of supply networks: a social network analysis approach. J. Oper. Manag. **29**, 194–211 (2011)
9. Rice, J., Caniato, F.: Building a secure and resilient supply network. Supply Chain Manag. Rev. **7**(5), 22–30 (2003)
10. Sawik, T.: Selection of resilient supply portfolio under disruption risks. Omega **41**, 259–269 (2013)
11. Spiegler, V., Naim, M., Wikner, J.: A control engineering approach to the assessment of supply chain resilience. Int. J. Prod. Res. **50**(21), 6162–6187 (2012)
12. Tsiakis, P., Shah, N., Pantelides, C.: Design of multi-echelon supply chain networks under demand uncertainty. Ind. Eng. Chem. Res. **40**(16), 3585–3604 (2001)

Exact Solution of Combined Cutting Stock and Scheduling Problems

Nuno Braga, Cláudio Alves, and José Valério de Carvalho

1 The Combined Cutting Stock and Scheduling Problem

Given their practical relevance and their challenging nature, cutting and packing problems have been a major topic of research since many years. The typology of Wäscher et al. [7] is a good illustration of this fact. Apart from providing a wide classification scheme for these problems, it also identifies a large set of contributions for different variants of the standard problem. These variants cover different characteristics of the items and rolls and different kinds of objective functions, for example. In this paper, we address one possible variant of the problem that considers a scheduling term in the objective function related to the existence of due dates imposed on the delivery of the items. This problem has been addressed recently by Reinertsen and Vossen in [6] and Arbib and Marinelli in [1].

The combined cutting stock and scheduling problem addressed in this paper is defined as follows. We are given a set I of one-dimensional items ($|I| = n$) with sizes denoted by w_i and a corresponding demand b_i, $i \in I$. These items have to be cut from stock rolls of standard length W whose availability is assumed to be unlimited. For bin-packing instances, the demands b_i, $i \in I$, are typically very low (near from 1), while cutting stock instances are characterized by high demands. Since the approaches devised in this paper are independent of the level of demands, we will now on refer only to the more general cutting stock case. The scheduling part of the problem is based on the fact that due dates are imposed on the delivery of each item size. As a consequence, for each item size $i \in I$, there is a corresponding due date which is denoted by d_i. From a scheduling standpoint, the item sizes can be seen as jobs whose delivery is expected at most until d_i units of time. Deliveries

N. Braga • C. Alves (✉) • J. Valério de Carvalho
Universidade do Minho, 4710-057 Braga, Portugal
e-mail: nuno.braga@dps.uminho.pt; claudio@dps.uminho.pt; vc@dps.uminho.pt

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_17

after the due dates are tolerated, but with a penalty in the objective function. Cutting one stock length is assumed to take exactly one unit of time. Hence, the time to complete the job related to an item size i corresponds to the number of stock rolls that have been cut up to the last one on which item i is cut. The objective function of the problem considers two terms: one related to the total wastage and another related to the total tardiness. Here, we will follow the approach of Arbib and Marinelli in [1], and consider that each term has the same weight in the objective function.

A similar problem has been explored by Li in [5] for the two-dimensional case with different stock lengths and jobs with different item sizes. The author considers the existence of both release and due dates, and she proposes linear programming (LP) based heuristics and non LP-based heuristics to find feasible solutions for the problem. As pointed out in [1], the models described in [5] do not ensure that an optimal solution is found for the global problem. The reason lies on the time available for each time period which defines the period length. In [6], the authors address the one-dimensional problem with due dates and no release dates. They describe an integer programming (IP) model which is an extension of the column generation model for cutting stock problems. The variables are associated to patterns indexed by time period. The planning horizon is partitioned into n distinct time periods. Then, the job related to an item size i can only be cut up to the i-th time period. Again, in [1], the authors showed that this approach may lead to sub-optimal solutions whenever an *early due date first*-based solution is not the optimal solution for the problem. The first exact formulation for this problem has been proposed in [1]. As in [6], the authors consider a formulation in which the variables are related to cutting patterns indexed by time period. Additionally, they use variables representing the inventory level of an item size at a given time period, and they use them in combination with equilibrium equations to identify situations of tardiness. To cope with the size of their model, the authors describe a period-splitting procedure which allows to start with larger period lengths. It consists in solving iteratively the model with shorter period lengths. The computational results provided by the authors illustrate the inherent difficulty of the problem.

In this paper, we explore a compact assignment formulation for the problem that applies to both bin-packing and cutting stock instances. We describe different inequalities to improve the quality of its continuous lower bound (Sect. 2), and we describe in particular an approach based on knapsack-based inequalities derived using dual-feasible functions (Sect. 3). To illustrate the potential of these approaches, we report on computational experiments performed on a set of benchmark instances (Sect. 4).

2 An Assignment Formulation

2.1 The Model

Let T denote the length of the planning horizon. The combined cutting stock and scheduling problem (with due dates) can be formulated with variables x_i^t, for each item $i = 1, \ldots, n$, and $t = 1, \ldots, T$, that represent the number of items of size i that are cut at time period t. The period length is assumed to be equal to 1, and hence, there is a one to one relation between time periods and the stock rolls that are cut. The binary variables z^t, $t = 1, \ldots, T$, determine wether a stock roll is used or not at time period t. The tardiness of an item size i, $i = 1, \ldots, n$, is measured through the binary variables y_i^t, $t = di + 1, \ldots, T$, which is defined only from the due date plus one time period until the end of the planning horizon. Variables y_i^t take the value 1 only if the item size i is cut on time period t. Hence, if $y_i^t = 1$ for some $i = 1, \ldots, n$, and $t = di + 1, \ldots, T$, the item size will be late by at least $t - d_i$ units of time. These variables may be related to the y_i^k variables used by Arbib and Marinelli in [1]. The assignment formulation states as follows.

$$\text{min.} \quad \sum_{t=1}^{T} z^t + \sum_{i=1}^{n} \sum_{t=d_i+1}^{T} y_i^t \tag{1}$$

$$\text{s.t.} \quad \sum_{t=1}^{T} x_i^t = b_i, \quad i = 1, \ldots, n, \tag{2}$$

$$\sum_{i=1}^{n} w_i x_i^t \leq W z^t, \quad i = 1, \ldots, n, \; t = 1, \ldots, T, \tag{3}$$

$$x_i^t + y_i^{t+1} \leq L_i^{min} y_i^t, \quad i = 1, \ldots, n, \; t = d_i + 1, \ldots, T, \tag{4}$$

$$z^t \in \{0, 1\}, \quad t = 1, \ldots, T, \tag{5}$$

$$x_i^t \geq 0 \text{ and integer}, \quad i = 1, \ldots, n, \; t = 1, \ldots, T, \tag{6}$$

$$y_i^t \in \{0, 1\}, \; i = 1, \ldots, n, \quad t = d_i + 1, \ldots, T. \tag{7}$$

As mentioned above, the objective function (1) is composed by a term related to the total wastage and another related to the total tardiness. Constraints (2) ensure the satisfaction of the demand. Constraints (3) forbid any cutting pattern that exceeds the stock length, while constraints (4) support the definition of the y_i^t variables. The values L_i^{min} for $i = 1, \ldots, n$, may be defined as follows:

$$L_i^{min} = \min \left\{ b_i, \left\lfloor \frac{W}{w_i} \right\rfloor + 1 \right\}.$$

2.2 Strengthening the Assignment Formulation

The LP relaxation of (1)–(7) can be strengthened by considering the following inequalities. Since a feasible solution will never use less than the stock rolls required to cut all the items, we may enforce the inequality

$$\sum_{t=1}^{T} z^t \geq K^{min}, \tag{8}$$

where K^{min} is a lower bound on the number of stock rolls computed for example using dual-feasible functions [3]. Assigning an item to a roll (or time period) implies that a stock roll is used at this time period, and hence, we have

$$z^t \geq y_i^t, \quad i = 1, \ldots, n, \ t = d_i + 1, \ldots, T. \tag{9}$$

Using a stock roll at time period $t + 1$, $t = 1, \ldots, T - 1$ implies that a roll has been used at time period t, and hence

$$z^t \geq z^{t+1}, \quad t = 1, \ldots, T. \tag{10}$$

Similar constraints apply to the y_i^t variables, which translate into the following inequalities

$$y_i^t \geq y_i^{t+1}, \quad t = d_i + 1, \ldots, T. \tag{11}$$

3 Knapsack-Based Inequalities Derived from Dual-Feasible Functions

3.1 Dual-Feasible Functions and Valid Inequalities

A function $f : [0, 1] \rightarrow [0, 1]$ is a *dual-feasible function (DFF)*, if for any finite set $\{x_i \in \mathbb{R}_+ : i \in J\}$ of nonnegative real numbers, the following holds

$$\sum_{i \in J} x_i \leq 1 \Longrightarrow \sum_{i \in J} f(x_i) \leq 1.$$

Dual-feasible functions have been used essentially to derive lower bounds for standard cutting and packing problems. In [3], the authors showed that these functions can also be used to derive valid inequalities for IP models with knapsack constraints.

To strengthen the formulation (1)–(7), we applied the DFF mentioned below to the knapsack constraints (3). Let f be a DFF. The following inequalities derived from (3) by applying f are valid inequalities for the integer polytope of (1)–(7):

$$\sum_{i=1}^{n} f\left(\frac{w_i}{W}\right) x_i^t \leq 1, \quad i = 1, \ldots, n, \ t = 1, \ldots, T. \tag{12}$$

Among all the DFF, the maximal functions are those that produce non-dominated results. In our experiments, we used the following functions that were shown to be maximal in [3]:$f_{FS,1}$ (described in [4]), with $k \in \mathbb{N} \setminus \{0\}$, and $f_{CCM,1}$ (described in [2]), with $C \in \mathbb{R}$ and $C \geq 1$.

- $f_{FS,1}(x; k) = \begin{cases} x, & \text{if } (k+1) * x \in \mathbb{N}, \\ \lfloor (k+1)x \rfloor / k, & \text{otherwise.} \end{cases}$
- $f_{CCM,1}(x; C) = \begin{cases} \lfloor Cx \rfloor / \lfloor C \rfloor, & \text{if } x < 1/2, \\ 1/2, & \text{if } x = 1/2, \\ 1 - f_{CCM,1}(1-x), & \text{if } x > 1/2. \end{cases}$

3.2 A Cutting Plane Procedure

From the definitions given above, we define the following general cutting plane procedure for (1)–(7).

Cutting plane procedure
Solve the LP relaxation of (1)–(7) (let (z^*, x^*, y^*) be the optimal solution);
do

Let f^k be a DFF with parameter γ;
for (*a given range of γ, and a given set of knapsack constraints (3)*) **do**
if (*inequality (12) is violated for f^γ, x^* and the current constraint (3)*)
then (*add this inequality (12) to the LP relaxation of (1)–(7)*)
end for.
Re-Solve the LP relaxation of (1)–(7) (let (z^*, x^*, y^*) be the optimal solution);

while (*stopping criteria are not satisfied*)**.**

To control the number of inequalities that are generated, we may limit the range of the DFF parameters and the number of knapsack constraints (3) from which the inequalities are derived. Furthermore, the stopping criteria may include for example the number of inequalities added to the LP relaxation or the maximum number of iterations without a significant improvement of the value of the objective function (based on a pre-defined value).

4 Computational Experiments

To evaluate the performance of the approaches devised in this paper, and in particular, the cutting plane procedure described in Sect. 3, we performed a set of computational experiments on benchmark instances. We used a set of instances taken from the same set of cutting stock instances as in [1] and for which due dates were generated, and we used values of T (the planning horizon) equal to those reported by these authors. A set of 40 instances divided in two groups of 20 instances were generated in that way. The tests were run on a PC with an i7 CPU with 3.5 GHz and 32 GB of RAM. For the optimization subroutines, we resorted to CPLEX 12.5.

In Table 1, we report on the values of the continuous lower bounds provided by the LP relaxation of (1)–(7) under the following scenarios:

- A : LP relaxation (1)–(7) without any cut;
- B : LP relaxation (1)–(7) with cut (8);
- C : same as B plus the cuts obtained with the cutting plane procedure of Sect. 3;
- D : same as C plus the cuts (9)–(11).

Columns $D1$ to $D4$ extend the previous results for different parameters of the cutting plane procedure discussed in Sect. 3. Let δ, max_{cuts}, and it denote respectively the smallest value of the violation of an inequality (12) that triggers its insertion on the model in the cutting plane procedure, the maximum number of cuts added between two consecutive resolutions of the LP relaxation, and the maximum number of iterations that we allow without an improvement of the solution value of the LP relaxation greater than 0.001. For the aforementioned scenarios C and D, the values for (δ, max_{cuts}, it) that were used are $(0.1, 20, 20)$. Columns $D1$ to $D4$ in Table 1 correspond to the previous scenario D with the following pairs of values for (δ, max_{cuts}, it):

- $D1$: $(\delta, max_{cuts}, it) = (0.1, 20, 50)$;
- $D2$: $(\delta, max_{cuts}, it) = (0.1, 50, 50)$;
- $D3$: $(\delta, max_{cuts}, it) = (0.01, 20, 20)$;
- $D4$: $(\delta, max_{cuts}, it) = (0.25, 20, 20)$.

Columns z_{RL} and t_{RL} in Table 1 stand respectively for the value of the optimal solution and the computing time required in the corresponding scenario. Note that we used the two dual-feasible functions described in Sect. 3 without any restriction on the domain of their parameters. Instead, we used the parameters (δ, max_{cuts}, it) to control the number of cuts that are added to the LP relaxation.

While inequality (8) leads to a small improvement in the value of the lower bound provided by the LP relaxation of (1)–(7), the impact of the cutting plane procedure described in Sect. 3 is much more significant. On average, the value of the lower bound increased by almost 4.7 % for the first set of instances, and by 11.1 % for the second set (scenario A against scenario C). Inequalities (9)–(11) have no impact on the first set of instances, but when combined with the cutting plane procedure, the average improvement of the lower bound raises to 15 %. That can be explained

Table 1 Computational results

			A		B		C		D		D1		D2		D3		D4	
Inst.	n	T	z_{RL}	t_{RL}	z_{RL}	t_{RL}	z_{RL}	t_{RL}	z_{RL}	t_{RL}	z_{RL}	t_{RL}	z_{RL}	t_{RL}	z_{RL}	t_{RL}	z_{RL}	t_{RL}
1	20	481	264.83	0.23	267.03	0.25	267.90	4.15	268.133	13.60	328.00	275.07	328.00	299.55	267.762	6.55	311.00	274.30
2	20	815	503.76	0.53	505.76	0.57	516.90	48.28	516.91	97.58	516.91	100.50	516.91	124.76	516.569	82.86	516.91	198.59
3	20	703	397.35	0.47	400.12	0.49	407.72	10.86	400.908	13.06	407.03	49.16	407.03	61.33	407.027	55.79	400.80	53.01
4	20	825	502.37	0.70	511.28	0.97	520.22	30.32	520.249	45.25	520.25	45.43	520.25	46.74	520.258	26.04	564.18	779.44
5	20	619	368.24	0.30	369.44	0.31	376.41	12.67	370.196	12.05	389.75	124.44	378.46	89.35	370.196	9.46	369.44	18.28
6	20	787	511.45	0.54	511.85	0.56	520.17	14.82	513.519	20.35	520.31	70.11	520.42	72.90	513.545	17.68	513.98	102.68
7	20	675	370.53	0.41	370.86	0.41	371.60	14.67	386.397	278.75	403.00	717.63	403.00	917.87	371.759	20.73	371.11	40.69
8	20	851	486.53	0.57	530.24	0.68	537.15	17.80	536.199	32.66	537.32	49.14	637.00	424.05	537.323	38.98	537.29	103.51
9	20	683	390.71	0.44	395.07	0.50	424.04	42.24	409.967	58.43	409.97	55.27	423.70	99.96	409.967	51.17	449.00	377.05
10	20	727	473.04	0.47	474.18	0.57	483.07	7.03	483.134	17.18	483.13	18.80	483.13	23.18	483.134	18.01	515.89	381.80
11	20	475	279.35	1.85	280.32	0.22	280.32	1.09	280.409	2.94	280.41	4.52	280.51	4.97	280.409	2.87	280.41	3.51
12	20	875	511.12	0.57	546.13	0.60	560.73	17.46	560.971	37.30	560.98	50.65	560.69	44.67	560.727	31.82	560.82	63.40
13	20	687	366.33	0.44	367.36	0.49	371.70	10.50	371.664	14.30	471.00	692.77	471.00	613.30	371.752	16.71	438.30	873.37
14	20	703	441.90	0.44	442.10	0.48	443.04	10.57	452.59	177.12	452.67	169.03	462.42	296.38	442.96	10.67	442.21	6.66
15	20	747	473.52	0.44	481.40	0.49	547.50	103.29	547.5	242.32	547.50	272.47	547.50	280.47	501.221	51.08	525.75	497.08
16	20	645	355.41	0.42	362.37	0.63	364.30	7.30	369.66	33.31	369.74	42.53	369.68	39.66	364.204	14.05	414.00	888.98
17	20	711	475.20	0.44	493.13	0.54	496.56	5.12	496.723	9.53	504.72	33.62	515.49	73.90	496.742	9.09	522.88	275.17
18	20	535	339.61	2.37	340.25	0.25	365.39	75.35	340.592	6.26	365.46	150.99	365.46	148.69	343.075	38.85	340.34	26.41
19	20	661	419.26	0.37	423.04	0.40	451.10	25.98	450.846	73.32	450.85	62.51	450.94	69.80	435.591	37.66	423.11	28.06
20	20	773	495.68	0.53	500.96	0.56	517.07	14.11	517.106	31.33	517.17	36.08	536.02	217.57	517.166	28.34	517.39	180.54

(continued)

Table 1 Continued

			A		B		C		D		D1		D2		D3		D4	
Inst.	*n*	*T*	z_{RL}	t_{RL}	z_{RL}	t_{RL}	z_{RL}	t_{RL}	z_{RL}	t_{RL}	z_{RL}	t_{RL}	z_{RL}	t_{RL}	z_{RL}	t_{RL}	z_{RL}	t_{RL}
average			421.31	0.63	428.64	0.50	441.14	23.68	439.68	60.83	451.81	151.04	458.88	197.45	435.57	28.42	450.74	258.63
21	20	481	290.51	0.31	292.71	0.33	338.60	28.28	337.67	29.70	350.93	70.67	350.93	227.76	337.67	44.70	343.79	144.28
22	20	815	528.57	0.40	530.57	0.44	582.00	10.13	590.11	42.00	607.83	165.07	607.84	201.02	587.73	29.67	589.91	145.16
23	20	703	441.44	0.29	444.21	0.29	501.35	5.47	536.83	68.07	537.07	119.97	537.08	278.43	484.22	15.63	537.04	191.49
24	20	825	545.66	0.43	554.58	0.47	645.00	13.92	579.75	22.68	659.19	214.13	659.19	883.62	587.89	50.36	659.19	287.84
25	20	619	387.29	0.33	388.49	0.36	431.51	27.24	442.65	70.25	442.65	66.47	442.65	192.47	443.28	126.16	431.88	71.46
26	20	787	544.06	0.35	544.45	0.35	633.00	10.66	651.86	68.03	651.92	85.62	651.86	262.19	651.86	256.39	651.86	67.80
27	20	675	407.52	0.36	407.84	0.41	439.50	37.43	447.12	93.20	447.12	85.33	447.12	183.44	448.52	240.55	445.45	49.00
28	20	851	494.68	0.42	538.39	0.51	543.02	6.87	568.76	18.84	683.16	463.53	683.22	817.41	568.50	18.58	669.82	231.24
29	20	683	419.91	0.34	424.27	0.42	469.95	4.92	529.93	78.09	529.93	85.13	529.93	272.00	529.93	269.44	529.93	106.40
30	20	727	538.88	0.35	540.01	0.38	603.00	6.32	613.84	49.56	613.86	81.28	613.86	249.34	613.84	78.63	613.84	95.99
31	20	475	303.46	0.24	304.42	0.26	320.03	6.02	329.24	24.08	329.24	22.54	329.24	68.63	330.69	64.04	329.11	21.54
32	20	875	581.92	0.33	616.93	0.46	715.00	13.37	736.32	105.90	736.32	121.21	736.32	370.26	648.76	27.87	736.32	81.51
33	20	687	427.85	0.26	428.88	0.29	445.15	3.05	536.77	81.78	536.77	71.00	536.77	213.52	536.77	322.55	536.72	51.73
34	20	703	471.20	0.32	471.40	0.35	489.10	16.62	503.76	15.16	531.32	76.08	531.32	216.78	531.55	142.37	503.47	37.80
35	20	747	496.21	0.42	504.09	0.46	509.68	3.82	581.83	131.04	581.83	122.02	581.66	510.64	584.13	462.79	580.15	156.05
36	20	645	409.30	0.39	416.26	0.37	468.25	8.49	475.79	57.32	475.79	67.06	475.79	255.54	475.64	134.07	475.79	161.29
37	20	711	498.86	0.41	516.78	0.45	525.87	8.07	555.42	18.77	611.17	170.04	610.86	623.89	555.50	13.68	561.52	66.63
38	20	535	356.18	0.33	356.82	0.36	400.90	23.31	381.81	16.19	408.37	141.29	408.37	243.16	381.19	11.71	399.64	62.48
39	20	661	449.39	0.33	453.17	0.36	465.30	4.09	511.60	87.38	511.60	230.48	511.60	325.48	498.75	26.20	505.83	97.91
40	20	773	536.03	0.34	541.31	0.38	614.00	8.84	625.24	71.83	625.24	287.96	635.24	246.81	571.30	14.15	635.24	257.57
average			456.45	0.35	463.78	0.38	507.01	12.35	526.81	57.49	543.56	137.34	544.04	332.12	518.39	117.47	536.82	119.26

by the fact that the DFF based cutting planes focus on the knapsack constraints of the model and particularly on the x_i^t variables, while inequalities (9)–(11) focus on the remaining variables of the model, thus complementing the cutting plane procedure. Considering the parameters (δ, max_{cuts}, it), the best results in terms of the quality of the continuous lower bounds are achieved with scenario $D2$ for both sets of instances. The corresponding average improvement in the value of the lower bound is respectively 8.9 % and 19.2 % for the first and second set of instances. The maximum improvement in the first set of instances goes up to 30.9 %, and up to 38.1 % for the second set.

Additionally, we solved the first set of instances up to integrality with a time limit of 3600 s for branch-and-bound in a way that is similar to the approach followed in [1]. For this purpose, we used the cutting plane procedure with the definitions of scenario D. After the time limit, the value of the best lower bound was better than the bound at the root node by 13.3 %, while the average integrality gap was equal to 20.4 %, a value that is in line with the overall gaps reported in [1] while obtained using a compact formulation.

Acknowledgements This work was supported by FEDER funding through the Programa Operacional Factores de Competitividade - COMPETE and by national funding through the Portuguese Science and Technology Foundation (FCT) in the scope of the project PTDC/EGE-GES/116676/2010.

References

1. Arbib, C., Marinelli, F.: On cutting stock with due dates. Omega **46**, 11–20 (2014)
2. Carlier, J., Clautiaux, F., Moukrim, A.: New reduction procedures and lower bounds for the two-dimensional bin-packing problem with fixed orientation. Comput. Oper. Res. **34**, 2223–2250 (2007)
3. Clautiaux, F., Alves, C., de Carvalho, J.M.V.: A survey of dual-feasible and superadditive functions. Ann. Oper. Res. **179**, 317–342 (2010)
4. Fekete, S., Schepers, J.: New classes of fast lower bounds for bin packing problems. Math. Program. **91**, 11–31 (2001)
5. Li, S.: Multi-job cutting stock problem with due dates and release dates. J. Oper. Res. Soc. **47**, 490–510 (1996)
6. Reinertsen, H., Vossen, T.: The one-dimensional cutting stock problem with due dates. Eur. J. Oper. Res. **201**, 701–711 (2010)
7. Wäscher, G., Haußner, H., Schumann, H.: An improved typology of cutting and packing problems. Eur. J. Oper. Res. **183**, 1109–1130 (2007)

Fair Transfer Prices of Global Supply Chains in the Process Industry

Songsong Liu, Roberto Fucarino, and Lazaros G. Papageorgiou

1 Introduction

A supply chain involves all activities transforming raw materials to final products and delivering them to the customers. During the past decade with rapid globalisation, many companies' production plants and delivery centres are located in multiple countries, maybe also in different continents. In a supply chain, usually its total profit aims to be maximised to enhance its performance. However, there is no automatic mechanism to allow profits to be fairly distributed among participants. Solutions with maximum total profit usually distribute profit quite unevenly, and are therefore impractical [9]. Transfer prices, consisting of procurement, manufacturing, and selling prices within a supply chain, can be used as a method to solve this problem. However, only a few papers have investigated the use of transfer price to distribute the profit fairly in the supply chains. Gjerdrum et al. proposed a mixed integer nonlinear programming (MINLP) model using Nash approach for the fair profit distribution in multi-enterprise supply chains [4], and later another MINLP model for fair transfer price and inventory holding policies in two-enterprise supply chains [5]. Chen et al. proposed a two-phase fuzzy decision-making method for a production and distribution planning model for a multi-echelon supply chain network to achieve multiple objectives such as maximising profit of

S. Liu (✉) • L.G. Papageorgiou (✉)
Centre for Process Systems Engineering, Department of Chemical Engineering, UCL (University College London), Torrington Place, London WC1E 7JE, UK
e-mail: s.liu@ucl.ac.uk; l.papageorgiou@ucl.ac.uk

R. Fucarino
Dipartimento di Ingegneria Chimica, Gestionale, Informatica, Meccanica, Università degli Studi di Palermo, Viale delle Scienze Ed., 8, 90128 Palermo, Italy
e-mail: r.fucarino@gmail.com

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_18

each participant enterprise, maximising customer service level, and ensuring fair profit distribution [3]. Then, the above work was extended to consider the demand and price uncertainties [2]. Rosenthal developed a cooperative game that provides transfer prices to allocate the net profit in a fair manner in both perfect information and asymmetric information cases [10]. Leng and Parlar constructed a cooperative game to fairly allocate firm-wide profit using transfer pricing for a two-echelon supply chain involving a single upstream division and multiple downstream [6].

This work addresses the production and distribution planning of the global supply chain with a fair profit distribution among the members using transfer prices. We aim to develop a mixed integer linear programming (MILP) optimisation framework, using the literature Nash and lexicographic maximin principles, to find the fair profit distribution to the supply chain's members. To the best of our knowledge, this work is the first one that applies the lexicographic maximin method to the supply chain profit distribution problem.

2 Problem Statement

The considered global supply chain network of an agrochemical company consists of several formulation plants and a number of market regions worldwide (Fig. 1). The products are categorised into different groups and each formulation plant can produce some specific product groups. The planning horizon is discretised into multiple time periods, and the demand of products varies among time periods. Final

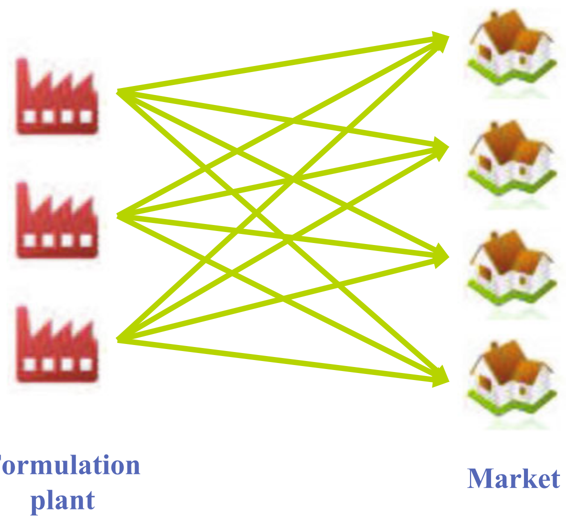


Fig. 1 Structure of a supply chain example

products are shipped from formulation plants to market regions, which occur the transportation cost and the duties cost. Capacities of formulation plants are constant during the planning horizon, as well as the selling prices of final products at the markets. The transfer prices are charged between formulation plants and markets.

In this problem, the given data include products, groups, formulation plants, market regions, weekly demands, capacities and capabilities of formulation plants, raw material costs, unit and fixed formulation costs, unit and fixed transportation costs, transportation times and duties from plants to markets, initial inventory and inventory limits, and selling prices. The decision variables include production of each formulation plant, transportation flows and transfer prices between formulation plants and markets, sales and lost sales at markets. The objective of the problem is to achieve a fair profit distribution among supply chain members.

3 Mathematical Formulation

The considered optimisation problem is formulated as an MILP model, extended from a literature model [7], which optimised the total cost of the whole supply chain as an objective, and did not consider the transfer prices between the supply chain members. In the proposed model, we aim to model the profit of each member using transfer prices as decision variables. In this section, we introduce the key constraints in the proposed model.

3.1 Production and Flow Constraints

If product i is produced at formulation plant j during time period t, i.e. binary variable $W_{ijt} = 1$, the production P_{ijt} is limited by minimum ($P_{ij}^{\min}$) and maximum ($P_{ij}^{\max}$) limits:

$$P_{ij}^{\min} \cdot W_{ijt} \leq P_{ijt} \leq P_{ij}^{\max} \cdot W_{ijt}, \quad \forall j, g \in G_j, i \in \bar{I}_g, t \tag{1}$$

where G_j indicates the set of product groups g that plant j can formulate, while $\bar{I}_g$ is the set of products that product group g involves.

If product i is shipped from formulation plant j to market k during time period t (binary variable $Y_{ijkt} = 1$), the shipped amount (F_{ijkt}) is limited by the minimum

($F_{ijk}^{\min}$) and the maximum ($F_{ijk}^{\max}$) limits:

$$F_{ijk}^{\min} \cdot Y_{ijkt} \leq F_{ijkt} \leq F_{ijk}^{\max} \cdot Y_{ijkt}, \quad \forall j, k, g \in G_j, i \in \bar{I}_g \cap I_k, t \tag{2}$$

where I_k refers to the set of products that are sold at market k.

3.2 Inventory Constraints

The inventory of product i at formulation plant j at the end of time period t (IV_{ijt}^F) is equal to the inventory of the product at the previous time period ($IV_{ij,t-1}^F$), plus the production in the time period (P_{ijt}), minus the total outgoing flows (F_{ijkt}):

$$IV_{ijt}^F = IV_{ij,t-1}^F + P_{ijt} - \sum_{k \in K_i} F_{ijkt}, \quad \forall j, g \in G_j, i \in \bar{I}_g, t \tag{3}$$

where K_i is the set of markets selling product i.

Similarly, the inventory of product i at market k at the end of time period t (IV_{ijt}^M) is equal to the total inventory of the product at the previous time period ($IV_{ij,t-1}^M$), plus any incoming flows (F_{ijkt}), minus sales (S_{ikt}):

$$IV_{ikt}^M = IV_{ik,t-1}^M + \sum_{j \in J_g} \sum_{g \in \bar{G}_i} F_{ijk,t-\tau_{jk}} - S_{ikt}, \quad \forall k, i \in I_k, t \tag{4}$$

where τ_{jk} is the transportation time from formulation plant j to market k; $\bar{G}_i$ is the set of product groups that product i belongs to; J_g is the set of formulation plants that can produce product group g.

3.3 Transfer Prices Constraints

Here, different from the literature work [7], transfer prices (TP_{ijk}) are considered as decision variables. Following [4], optimal transfer prices are selected from a set of candidate price levels ($\overline{TP}_{ijkm}$), and binary variable O_{ijkm} is introduced to indicate whether price level m is chosen:

$$TP_{ijk} = \sum_m \overline{TP}_{ijkm} \cdot O_{ijkm}, \quad \forall j, k, g \in G_j, i \in \bar{I}_g \cap I_k \tag{5}$$

Only one transfer price level can be chosen for each product between a formulation plant and a market:

$$\sum_{m} O_{ijkm} = 1, \quad \forall j, k, g \in G_j, i \in \bar{I}_g \cap I_k \tag{6}$$

3.4 Lost Sales Constraints

The sales (S_{ikt}) of product i at market k during time period t should not exceed its corresponding demand (D_{ikt}), while the unsatisfied amount is lost (LS_{ikt}):

$$D_{ikt} - S_{ikt} = LS_{ikt}, \quad \forall k, i \in I_k, t \tag{7}$$

3.5 Profit Constraints

At a formulation plant, its revenue (Re_j^F) is equal to the summation of the transfer price of a product multiplied by the corresponding flow between formulation plant and market:

$$Re_j^F = \sum_{t} \sum_{g \in G_j} \sum_{k} \sum_{i \in \bar{I}_g \cap I_k} TP_{ijk} \cdot F_{ijkt}, \quad \forall j \tag{8}$$

which is a nonlinear equation, and can be linearised using auxiliary variables and constraints, based on Eq. (5). The costs incurred for formulation plants include the raw material cost, formulation cost, inventory cost, and transportation cost. The profit of a formulation plant (Pr_j^F) is equal to its revenue minus its total cost (C_j^F).

$$Pr_j^F = Re_j^F - C_j^F, \quad \forall j \tag{9}$$

The revenue of each market (Re_k^M) is constituted by the selling prices of each product i at the market k (V_{ik}) multiplied by the sales of the product at each market:

$$Re_k^M = \sum_{t} \sum_{i \in I_k} V_{ik} \cdot S_{ikt}, \quad \forall k \tag{10}$$

While its costs include the purchase cost from formulation plants, inventory cost, duties, and lost sales cost. The profit of a market (Pr_k^M) is equal to its revenue minus its total cost (C_k^M):

$$Pr_k^M = Re_k^M - C_k^M, \quad \forall k \tag{11}$$

4 Solution Approaches

In this section, we apply two literature solution approaches for fair solutions, Nash approach and lexicographic maximin approach, to the considered optimisation problem.

4.1 Nash Approach

In the Nash approach, each member of the supply chain has a minimum acceptable profit, and the objective function of the model is given by the product of the difference between each member profit value and the corresponding lower bound:

$$\Psi = \prod_{n}(Pr_n - Pr_n^{\min}) \tag{12}$$

where Pr_n is the profit of member n in the supply chain, including formulation plants and markets, and $Pr_n^{\min}$ is the minimum acceptable value of each member's profit. The literature separable programming approach [4] is implemented to linearise to the nonlinear equation (12). Thus, the resulting model is of MILP format.

4.2 Lexicographic Maximin Approach

For a multiobjective maximisation problem, when each objective function is equally important as the concept of fairness requires, the lexicographic maximin approach can be applied here:

$$\text{Lexmax}_{\boldsymbol{x}\in\Omega}\Theta(\overline{Pr}_n(\boldsymbol{x})) \tag{13}$$

where $\overline{Pr}_n(\boldsymbol{x})$ is the normalised profit of member n in the supply chain. $\overline{Pr}_n$ can be calculated as follows:

$$\overline{Pr}_n = \frac{Pr_n - Pr_n^{\min}}{Pr_n^{\max} - Pr_n^{\min}}, \quad \forall n \tag{14}$$

In Eq. (14), $\Theta : \Re^N \rightarrow \Re^N$ is a mapping function that nondecreasingly orders the components of vector. Given a vector $\boldsymbol{e} = (e_1, \ldots, e_N)$, $\Theta(\boldsymbol{e}) = (\theta_1(\boldsymbol{e}), \ldots, \theta_N(\boldsymbol{e}))$, where $\theta_n(\boldsymbol{e}) \in \{e_1, \ldots, e_N\}$ is the nth component of the $\Theta(\boldsymbol{e})$ and $\theta_1(\boldsymbol{e}) \leq \ldots \leq \theta_N(\boldsymbol{e})$. $Pr_n^{\max}$ is the maximum value of each member's profit. In the lexicographic maximin problem, we maximise first the worst objective value first, then maximise the second worst objective value, the third worst value, and so on. The lexicographic maximin problem of Eq. (13) is transformed as a lexicographic maximisation problem [8], involving a set of MILP models to be solved iteratively.

5 Illustrative Example

In the illustrative example, the supply chain consists of three formulation plants (F1–F3) and four market regions (R1–R4) as shown in Fig. 1. There are eight products (P1–P8) in two product groups (G1–G2). Weekly demands in planning horizon of 8 weeks are known. Here, we investigate three scenarios, A, B and C:

- Scenario A: maximisation of the total profit of all members in the supply chain;
- Scenario B: fair profit distribution problem using Nash approach;
- Scenario C: fair profit distribution problem using lexicographic maximin approach.

In Scenarios B and C, the maximum profit of each member, $Pr_n^{\max}$, is obtained by maximising the profit of the single member. Then, 10 % of the achieved maximum profit is considered as the minimum acceptable profit, $Pr_n^{\min}$.

The developed models and approaches are implemented in GAMS [1] using CPLEX as MILP solver. The optimality gap was set to 0.1 %. The model statistics and computational performance of each scenario are presented in Table 1. It shows that the total supply chain profit is not significantly affected by considering the fair profit distribution, only about 2 % lower than the optimal value.

The profit of each member in the supply chain under the three scenarios is shown in Fig. 2. Table 2 compares the objective function terms in the Nash approach and lexicographic maximin approach for each member. Considering $Pr_n^* - Pr_n^{\min}$,

Table 1 Model statistics and computational performance of all scenarios

Scenario	No of equations	No of con variables	No of bin variables	CPU (s)	Total profit (rmu)
A	5666	4346	904	0.8	206,996
B	5750	4416	904	4.9	202,827
C	5735	4416	904	110.6	202,497

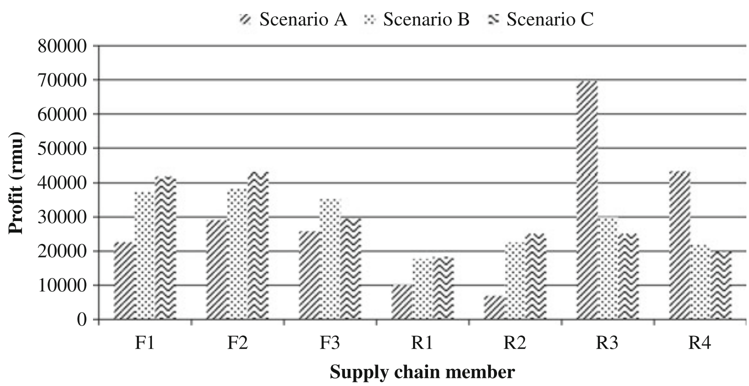


Fig. 2 Optimal profit distributions under all scenarios

Table 2 Comparison of profit distribution fairness

	Pr_n^{max}	Pr_n^{min}	$Pr_n^* - Pr_n^{min}$		$\overline{Pr}_n^*$	
			Scenario A	Scenario B	Scenario A	Scenario C
F1	124,603	12,460	10,129	24,930	0.090	0.262
F2	128,401	12,840	16,195	25,282	0.140	0.261
F3	87,962	8796	16,969	26,390	0.214	0.262
R1	54,059	5406	4677	12,267	0.096	0.263
R2	74,830	7483	−678	14,967	−0.010	0.262
R3	74,830	7483	62,049	22,450	0.921	0.262
R4	55,180	5518	37,629	16,555	0.758	0.289

its value of each member under Scenario B ranges between 12,000 and 27,000, while under Scenario A, the values fluctuate much more significantly. Thus, under Scenario B, all members have a more similar difference from their minimum profit requirement. Comparing $\overline{Pr}_n^*$, under Scenario C, all members have similar values between 0.261 and 0.263, except R4, while under Scenario A, R3's value is close to 1, but R2's value is negative. Thus, Scenario C finds a profit distribution that all members have similar relative profits based on the bounds. Although the distributions of total profit are different under Scenarios B and C, due to their different definitions of fairness, both approaches are able to find alternative fair solutions.

6 Concluding Remarks

In this work, an MILP model has been developed concerning the optimal production and distribution planning of a global supply chain in the process industry. Transfer prices have been used as a mechanism to fairly distribute the whole supply chain's profit. Two solution approaches, Nash approach and lexicographic maximin approach, have been used to find the fair profit distribution among the supply chain. The results of an illustrative example show that both methods can find fair solutions, and can be used as alternatives.

Acknowledgements Funding from the UK Engineering & Physical Sciences Research Council (EPSRC) for the EPSRC Centre for Innovative Manufacturing in Emergent Macromolecular Therapies hosted by University College London is gratefully acknowledged. Financial support from the consortium of industrial and governmental users is also acknowledged.

References

1. Brooke, A., Kendrick, D., Meeraus, A., Raman, R.: GAMS – A User's Guide. GAMS Development Corporation, Washington, DC (2012)

2. Chen, C.L., Lee, W.C.: Multi-objective optimization of multi-echelon supply chain networks with uncertain product demands and prices. Comput. Chem. Eng. **28**, 1131–1144 (2003)
3. Chen, C.L., Wang, B.W., Lee, W.C.: Multiobjective optimization for a multienterprise supply chain network. Ind. Eng. Chem. Res. **42**, 1879–1889 (2003)
4. Gjerdrum, J., Shah, N., Papageorgiou, L.G.: Transfer prices for multienterprise supply chain optimization. Ind. Eng. Chem. Res. **40**, 1650–1660 (2001)
5. Gjerdrum, J., Shah, N., Papageorgiou, L.G.: Fair transfer price and inventory holding policies in two-enterprise supply chains. Eur. J. Oper. Res. **143**, 582–599 (2002)
6. Leng, M., Parlar, M.: Transfer pricing in a multidivisional firm: a cooperative game analysis. Oper. Res. Lett. **40**, 364–369 (2008)
7. Liu, S., Papageorgiou, L.G.: Multiobjective optimisation of production, distribution and capacity planning of global supply chains in the process industry. Omega **41**, 369–382 (2008)
8. Ogryczak, W., Piòro, M., Tomaszewski, A.: Telecommunications network design and max-min optimization problem. J. Telecommun. Inf. Technol. **3**, 43–56 (2005)
9. Papageorgiou, L.G.: Supply chain optimisation for the process industries: advances and opportunities. Comput. Chem. Eng. **33**, 1931–1938 (2009)
10. Rosenthal, E.C.: A game-theoretic approach to transfer pricing in a vertically integrated supply chain. Int. J. Prod. Econ. **115**, 542–552 (2008)

The Influence of Corporate Social Responsibility on Economic Performance Within Supply Chain Planning

Bruna Mota, Maria Isabel Gomes, Ana Carvalho, and Ana Paula Barbosa-Póvoa

1 Introduction

There is a growing concern from customers about sustainability issues and governments are pressuring companies to become more sustainable and achieve a sustainable development, contributing towards the society goal of meeting the needs of the present without compromising the ability of future generations to meet their own needs, as defined by the Brundtland Commission [1]. If previously the concept of sustainability was more environmentally focused, currently the concept of the Triple Bottom Line is well established and sustainability is considered to be supported by three main pillars: economic, environmental and social sustainability [2]. However, while significant literature exists regarding the economic and environmental pillars, the social aspect of sustainability remains unaccounted for [4]. New legislation and standards are being introduced in an attempt to fill this gap in industry. ISO 26000:2010 for example, provides guidance to all types of organizations, and is intended to encourage them to go beyond legal compliance in the field of social responsibility. The European Commission itself has demonstrated concerns on this matter and has recently released the agenda for the 2014–2020 funding period, where the main objective is to fund projects contributing to regional

B. Mota (✉) • A. Carvalho • A.P. Barbosa-Póvoa
CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, Av. Rovisco Pais, 1049-001 Lisboa, Portugal
e-mail: bruna.mota@tecnico.ulisboa.pt; anacarvalho@tecnico.ulisboa.pt; apovoa@tecnico.ulisboa.pt

M.I. Gomes
Centro de Matematica e Aplicações, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Monte de Caparica, 2829-516 Caparica, Portugal
e-mail: mirg@fct.unl.pt

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_19

development and job creation [3]. All of these aspects are leading companies to look for ways to move towards a more sustainable state. Hence, there is a clear need for research in this field and tools to evaluate the impacts such strategies would have on the company's performance. This work takes a step forward in this direction and aims to provide a methodology for companies to study how they could in fact go beyond legal compliance, namely towards the contribution to regional development, and what would be the economic impact of such actions. A mathematical programming model is developed for the design and planning of supply chains, along with a social benefit indicator that promotes the economic growth of the less developed regions. The model is applied to a case study where the supply chain of a Portuguese battery producer is optimized under different objectives: an economic and a social one.

2 Problem Definition

The problem addressed here aims at determining the supply chain structure and the planning decisions that minimize costs and the ones that maximize social benefit. The problem is modeled through a Mixed Integer Linear Programming (MILP) that uses a graph approach for the design and planning of closed loop supply chains. The decisions at the design level are taken for a given time horizon (e.g. 5 years). This time horizon is divided in time periods (e.g. months) in which demand and return values must be satisfied. Detailed planning on attaining this satisfaction is given by the model. The economic performance of the supply chain is measured through the costs involved, as described in Eq. (1),

$$\text{Cost} = \sum_{i \in I} cf_i Y_i + \sum_{mij:(m,i,j) \in F_s} \sum_{t \in T} cs_{mit} X_{miit} + \sum_{ij:(i,j) \in A_{own}} \sum_{t \in T} ct_{ij} d_{ij} Z_{ijt} + \sum_{mij:(m,i,j) \in F_{out}} \sum_{t \in T} ct_{ij} d_{ij} X_{mijt} + \sum_{mi:(m,i) \in V_c} cp_{mi} \left(\sum_{j \in I} \sum_{t \in T} X_{mijt}\right) + \sum_{i \in I} chr_i Y_i. \tag{1}$$

The first term gives the fixed costs of each entity (cf_i) controlled by the binary variable Y_i which equals 1 when entity i is opened. The second term corresponds to the costs of raw materials where cs_{mit} represents the unit cost of product m acquired in entity i for period t, and X_{mijt} is a continuous variable for the amount of product m served by entity i to entity j at time t. The third term concerns the costs of transportation which is performed by the company's fleet, and depends on parameters such as vehicle consumption, fuel price and vehicle maintenance. The fourth term is related to outsourced transportation, which varies with contracted costs (per kg km), the amount of units transported and the kilometres travelled. The fifth term represents the costs of product recovery (cp_{mi}). The final term concerns the costs with human resources (chr_i) that result from opening a given entity. Regarding the social component, a social benefit indicator was developed that when maximized favours entities to be located in less developed regions, as described in Eq. (2),

$$\mathrm{SB} = \sum_{i \in I} u_i w_i Y_i, \tag{2}$$

where u_i represents a regional factor attributing a higher score to less developed regions (measured through regional statistics of population density), and w_i is the number of jobs created at region i. Additionally to the objective functions a set of constraints is defined which describes the structural and tactical decisions that need to be accounted for to satisfy certain market levels. Shortly, given: (a) a possible superstructure for the location of the supply chain entities, the associated investment costs as well as involved processes; (b) inventory, return and transportation policies and associated costs; (c) human resources costs and the social benefit associated to each facility; and (d) costumers demands; the goal is to determine (1) the network structure, (2) the production and storage levels, and (3) the flow amounts; so as to minimize the total supply chain cost and maximize the social benefit.

3 Case Study

The model was applied to a case study of a Portuguese lead battery manufacturer and distributor. This supply chain is composed by a factory in Oeiras (which is not to be relocated) and 12 rented warehouses in continental Portugal, serving around 2300 customers. The factory also has a storage function. Given the strategic nature of this work, customers were clustered in 237 groups, according to their municipality. Two hundred and thirty seven possible warehouse locations are then considered. A maximum number of 13 warehouses was imposed, according to the company's strategy. The company has a recycling strategy implemented for end-of-life batteries and thus there are both forward and reverse flows. Regarding distribution, the inbound transportation is outsourced, while the outbound transportation is performed by the company's fleet. All demand is to be fully satisfied. Three different scenarios were studied. Figure 1 shows the network obtained for each of the considered scenarios: the base case (A), the minimum cost solution (B) and the maximum social benefit solution (C). Scenario A shows the 13 warehouses that form the current supply chain of the company. Scenario B shows that only seven warehouses are necessary to achieve the minimum cost solution. In scenario C, 13 warehouses are opened and located in the less populated regions (darker regions indicate higher population density). These results are supported by the cost distribution of each scenario, represented in Fig. 2 (on the left), where we see that when minimizing cost (scenario B) the model returns a solution with a cost reduction of 21.5 %. This is obtained by reducing the number of warehouses, compensated by the increase in transportation. With scenario C the cost increases by 44 % (compared to scenario B), where an increase in both the number of warehouses and in transportation can be observed. The transportation costs for each scenario, represented in Fig. 2 (on

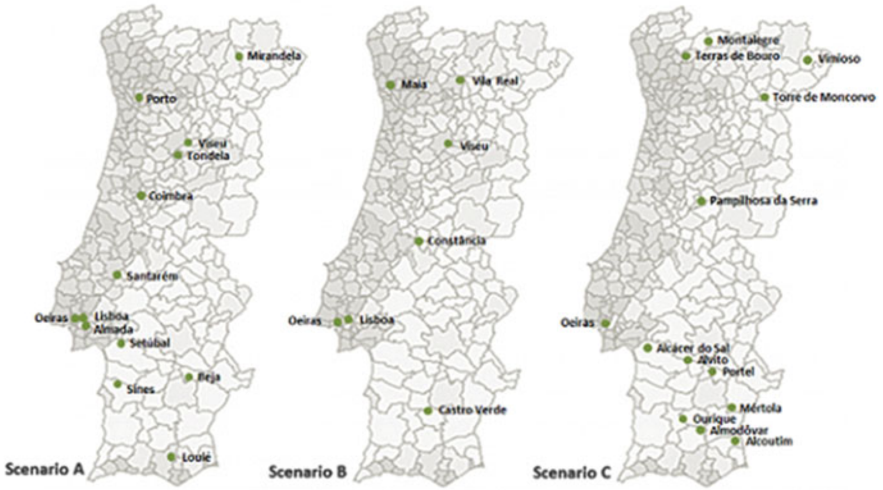


Fig. 1 Location of warehouses for each scenario: (A) base case, (B) minimizing cost, and (C) maximizing social benefit. *Darker regions* indicate higher population density

Fig. 2 Total cost (on the *left*) and transportation costs (on the *right*) distribution for each scenario

the right), are proportional to the increase in the secondary distribution. When looking with detail into the supply planning obtained for each of the scenarios it can be seen that, in scenario A (results not shown), Lisboa supplies 19 % of the products, followed by Setúbal (16 %) and Porto (12 %). In fact the clients with higher demand are located around these exact regions. In the minimum cost scenario (Fig. 3) it can be seen that the warehouse in Lisboa is responsible for the major share of supply (more than 40 %), since it is closer to the higher demand customers in Setúbal. It is interesting to see that the model preferably uses this warehouse, even though having the warehouse in Oeiras available with no costs of primary transport. Viseu and Maia supply the North, and Castro Verde supplies the southern municipalities. Oeiras, Constância and Vila Real play a more secondary role, complementing the mentioned warehouses. Finally, and having in mind the objective of maximizing the social benefit as a goal to pursue within a corporate social responsibility perspective, the optimal warehouse locations determined for maximum social benefit (scenario C) are imposed on the model. This network is then studied so as to determine the minimum cost supply planning. Figure 4 shows the results obtained. Lisboa and Porto are very densely populated regions. Therefore,

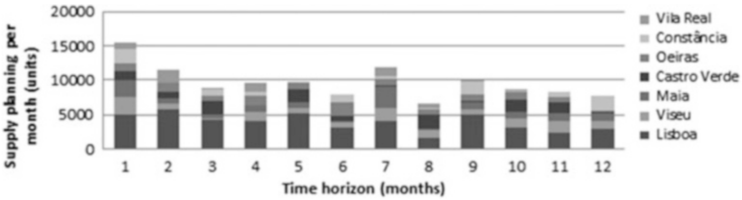


Fig. 3 Supply planning, from the factory to each warehouse, per month, for scenario B

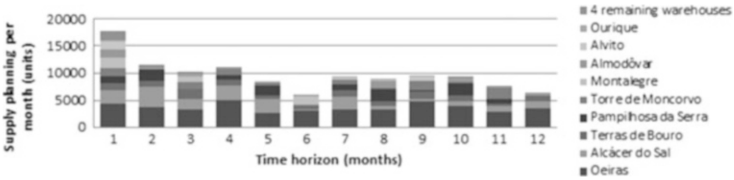


Fig. 4 Supply planning, from the factory to each warehouse, per month, for scenario C

when looking at the results of scenario C we see that these warehouses are replaced and the corresponding demand is reallocated. In this case the importance of Oeiras, the warehouse within the factory, is increased when compared to the minimum cost solution. Since the warehouses are so distant from the main clients, the model mostly uses the warehouse in Oeiras (38 % of the supply) to minimize the costs of primary distribution. Alcácer do Sal supports Oeiras supplying the region of Lisboa and Setúbal. Terras de Bouro, Torre de Moncorvo and Montalegre replace Maia and Vila Real in the supply of the northern clients. Pampilhosa da Serra covers the central regions. Having the satisfaction of the demand as a defined strategic goal—forward flow, it is important to look at how the obtained networks perform on the reverse flows (results not shown). Whereas in scenario A, Lisboa collects 19 % of the total end-of-life products, followed by Setúbal (16 %) and Porto (12 %), in scenario B Lisboa collects 41 % of the total end-of-life products and the warehouse co-located with the factory (Oeiras) collects around 12 %. In scenario C this warehouse gains importance once again and is responsible for about 52 % of battery recovery. One aspect that stands out in scenario C is that in several months, some warehouses are idle. This happens because the structure obtained in this case does not support a profitable planning as the demand is located far away from these warehouses. This result opens way to further research on how to improve this result, achieving a less expensive solution while still increasing the social benefit. Also worth mentioning is that very few units are supplied in some months from certain warehouses. This happens due to the constraint of having all demand satisfied, even if only one unit is to be supplied. In conclusion, although the pursue of a social benefit under a corporate social responsibility perspective should be targeted, it is important to have in mind if this can be applied in real terms in the organization. In the present case the analysis shows that further studies are required and it calls for a multiobjective approach to find a compromise solution amongst the three pillars of sustainability.

4 Conclusions and Future Work

This work proposes a mathematical optimization model for the design and planning of closed loop supply chains that serves as a tool to study the economic impact of incorporating corporate social responsibility issues in the company strategy. The model is applied to a case study where the promotion of regional development is intended. The results show that this strategy alteration translates in a quite different supply plan, accompanied by a compromise of the economic performance. However, room for improvement is identified. As future work, the social benefit indicator should be further refined to incorporate other regional development measures such as unemployment rate or GDP. The model should also be further generalized to minimize results with idle warehouses in the planning horizon, by adding new constraints. Also, as mentioned above, a multiobjective approach should be explored to find a compromise solution between the two objectives within a sustainable development perspective.

Acknowledgements The authors acknowledge the Portuguese National Science Foundation (FCT) for Grants SFRH/BD/51947/2012 and SFRH/BPD/63668/2009, and Projects PTDC/SEN-ENR/102869/2008, PEst-OE/MAT/UI0297/2011 and PTDC/EMS-SIS/1982/2012.

References

1. Brundtland, G.: Our Common Future - Report of the World Commission on Environment and Development (1987)
2. Elkington, J.: Enter the triple bottom line. In: Henriques, A., Richardson, J. (eds.) The Triple Bottom Line: Does It All Add Up, pp. 1–16. Taylor & Francis, London (2004)
3. European Commission: EU funding instruments 2014–2020 (2010)
4. Seuring, S.: A review of modeling approaches for sustainable supply chain management. Decis. Support Syst. **54**(4), 1513–1520 (2013)

A MIP Model for Production Planning in the Roasting Coffee Industry

Diana Yomali Ospina, Maria Antónia Carravilla, and José Fernando Oliveira

1 Introduction

Coffee is one of the most valuable primary products in the world and has a big impact in the economy and even in the politics of the developing countries. Brazil, Vietnam and Colombia are the main producers and for which coffee cultivation is an important source of income [5, 7]. To link the producers with the customers there is a need for a supply chain. A supply chain starts with unprocessed raw materials and ends with the final customer that uses the finished goods through a sequence of different activities, operations and resources depending on the final product. These processes must operate appropriately and provide accurate information to obtain the best performance [1, 3, 12].

Typically, the coffee supply chain includes four main stages: harvesting, green coffee commercialization, production, and distribution [11]. The harvesting phase includes growing and coffee beans treatment and the result is green coffee. The green coffee can be commercialized for the roasted coffee market. The production phase includes storage, roasting, grinding, blending and packaging which are carried out in order to meet different requirements in terms of freshness, aroma, flavour and color of coffee drinks. Finally the coffee can be distributed and is ready for consumption [4, 6, 13].

M.A. Carravilla • J.F. Oliveira
INESC TEC, Faculty of Engineering, University of Porto, 4200-465 Porto, Portugal
e-mail: mac@fe.up.pt; jfo@fe.up.pt

D.Y. Ospina (✉)
Department of Production and Mechanical Engineering, Universidad Autónoma de Manizales, Manizales, Colombia
e-mail: dianaospina@autonoma.edu.co

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_20

This paper presents a case study of a leading Portuguese roasting coffee company. A MIP model of the production phase has been built to be embedded in a Decision Support System (DSS) for the production planning of the coffee company. This model enables the daily planner to test the acceptance of new orders and at the beginning of each day determines the silo loads and the production quantities in order to produce as near as possible to the due date to preserve the product freshness.

The paper is organized as follows. The process of roasting and grinding coffee in general and the specific process in the company are described in Sect. 2. The production planning model for the case study is presented in Sect. 3. Section 4, details some insights on the results obtained with the model and Sect. 5 summarizes the study main findings and points out some future work.

2 The Coffee Production Process

An important phase in the coffee supply chain is the production. Some authors have presented works with applications in the roasted coffee process [2, 4, 8, 9]. Already in 1881 [10] it had been presented a thorough description of the whole supply chain and specifically on the production process from green coffee to roasting and grinding coffee.

As stated by Hicks [4] coffee from various origins are blended in different proportions to obtain predefined characteristics in terms of aroma, flavour and acidity. Roasting, usually the second phase, is considered the most important phase for flavours development. There are two most common methods to roast coffee: drum roasting and hot air roasting. The roasting phase is done in batches, small batches in case of speciality coffee. During the roasting process the beans increase in size (50–80 %) and lose weight (approx. 16 %) at the same time. The roasted beans may be cooled in three different ways, with water, and with normal or forced air. The coffee beans may be directly packed or the coffee may be ground and then packed.

Every morning our case study company follows the production process by feeding each silo with green coffee from a different origin in a predefined quantity that will be used during the day to build the blends that will be produced. Blends usually contain between three to five different types of beans. In this company the blending is followed by drum roasting. After being roasted, the beans are cooled with forced air and are transported either directly to the packaging machines or to the grinding machines to be subsequently packed as ground coffee. The ground coffee may be packed in sachets or in bags.

3 MIP Model for Production Planning

The main interest of the company is to have a production planning model embedded in a DSS that allows them to test at the beginning of each day, the implications of the acceptance of orders for a given due date. The system should therefore provide answers very quickly in order for them to make several tests at the beginning of the morning to define the production quantities for each day and the silo loading.

3.1 Indices

$$
\begin{aligned}
e \in \mathscr{E}, &\quad \mathscr{E} = \{1, \ldots, E\} \text{ — order;}\\
t \in \mathscr{T}, &\quad \mathscr{T} = \{1, \ldots, T\} \text{ — period;}\\
s \in \mathscr{S}, &\quad \mathscr{S} = \{1, \ldots, S\} \text{ — silo;}\\
p \in \mathscr{P}, &\quad \mathscr{P} = \{1, \ldots, P\} \text{ — product;}\\
c \in \mathscr{C}, &\quad \mathscr{C} = \{1, \ldots, C\} \text{ — type of coffee.}
\end{aligned}
$$

3.2 Data

Capacities:

CS_s	Capacity of silo s by period (kg);
CT	Roasting capacity by period (kg);
CEG	Bean coffee packing capacity by period (kg);
CM	Grinding capacity by period (kg);
CEM_1	Ground coffee packing type 1 capacity by period (kg);
CEM_2	Ground coffee packing type 2 capacity by period (kg).

Yields:

RT	Roasting yield (≤ 1);
RM	Grinding yield (≤ 1);
REG	Bean coffee packing yield (≤ 1);

Products (p):

T_p	Processing time for product p in roasting per (kg);
B_{pc}	Quantity of coffee type c that 1 kg of product p contains.

Orders (e):

$P_e \in \mathscr{P}$	The product of order e;
Q_e	Quantity of order e (kg);
$D_e \in \mathscr{T}$	Delivery date of order e;
A_e	Finishing of order e (bean, ground).

Where:

$A_e = 11$,	Final product in bean, 1, and packaging in bags, 1;
$A_e = 12$,	Final product in bean, 1, and packaging in sachets, 2;
$A_e = 21$,	Final product ground, 2, and packaging in bags, 1;
$A_e = 22$,	Final product ground, 2, and packaging in sachets, 2.

3.3 Decision Variables

x_{et} Quantity of order e to produce in period t (kg);

$$\delta_{sct} = \begin{cases} 1 & \text{if the silo } s \text{ is loaded with coffee type } c \text{ in period } t \\ 0 & \text{otherwise} \end{cases}$$

CA_{ct} Sum of capacities of the silos with coffee type c in period t (kg);

3.4 Objective Function

$$\min \sum_{e,\ t<D_e} x_{et}(D_e - t) + \sum_{e,\ t=D_e+1} 2x_{et} + \sum_{e,\ t>D_e+1} x_{et}(t - D_e)^2 + \sum_{s,c,t} \delta_{sct}$$

The objective function has two components. The first one aims to ensure that each order e is produced as close as possible to its delivery date (D_e). The production in the period $t < D_e$ implies a storage cost, proportional to the number of periods and to the quantity stored. If the product is delivered on D_e, the storage cost is zero. If the delivery is in period $t = D_e + 1$, a delay cost of two times the quantity ordered will be assigned. For $t > D_e + 1$ the cost will be the delayed quantity multiplied by the square of the delay, $(t - D_e)^2$. The second component aims to minimize the number of silos that are filled with coffee at the beginning of each period.

3.5 Constraints

$$\sum_{e \in \mathscr{E}: P_e = p} B_{pc} x_{et} \leq CA_{ct} \quad \forall t, c \tag{1}$$

$$\sum_{s \in \mathscr{S}} CS_s \delta_{sct} = CA_{ct} \quad \forall t, c \tag{2}$$

$$\sum_{c \in \mathscr{C}} \delta_{sct} \quad \leq 1 \quad \forall t, s \tag{3}$$

$$\sum_{e \in \mathscr{E}: P_e = p} T_p x_{et} \quad \leq CT \quad \forall t \tag{4}$$

$$\sum_{e \in \mathscr{E}: A_e = \{21;22\}} RT \times x_{et} \quad \leq CM \quad \forall t \tag{5}$$

$$\sum_{e \in \mathscr{E}: A_e = \{11;12\}} RT \times x_{et} \quad \leq CEG \quad \forall t \tag{6}$$

$$\sum_{e \in \mathscr{E}: A_e = \{21\}} RT \times RM \times x_{et} \leq CEM_1 \; \forall t \tag{7}$$

$$\sum_{e \in \mathscr{E}: A_e = \{22\}} RT \times RM \times x_{et} \leq CEM_2 \; \forall t \tag{8}$$

$$\sum_{t \in \mathscr{T}} RT \times RM \times x_{et} \quad \geq Q_e \quad \forall e \in \mathscr{E} : A_e \in \{21; 22\} \tag{9}$$

$$\sum_{t \in \mathscr{T}} RT \times x_{et} \quad \geq Q_e \quad \forall e \in \mathscr{E} : A_e \in \{11; 12\} \tag{10}$$

Constraints (1) ensure that, for each period t and coffee type c, the quantity of coffee type c required to produce the orders is less than or equal to the amount of coffee type c that exist in all silos (CA_{ct}). Constraints (2) determine, for each period t and coffee type c, the sum of the silo capacities which have been assigned to coffee type c (CA_{ct}). Constraints (3) ensure that, for each period t, one silo can only contain at most one type of coffee c. Constraints (4) ensure that, for each period t, the roasting capacity is not exceeded. Constraints (5) ensure that, for each period t, there exits grinding capacity to produce orders of ground coffee. Constraints (6) ensure that, for each period t, the quantity of coffee beans to be packed does not exceed the packaging capacity of coffee beans. Constraints (7) ensure that, for each period t, the quantity of ground coffee to be packed in bags does not exceed the packaging capacity of ground coffee in bags. Constraints (8) ensure that, for each period t, the quantity of ground coffee to be packed in sachets does not exceed the packaging capacity of ground coffee in sachets. Constraints (9) and (10) ensure that, for the planning horizon and for each order e, the quantity to be produced is equal to or greater than the quantity ordered.

Table 1 Computational results

Instance	1	2	3	4	5
Quantity of order	2000	2000	2000	2500	3000
Objective function	82,028	10,030	12,040	18,040	426,038
GAP (%)	0	0	0	0	0
Iterations	12,729	12,186	74,723	127,240	630
Nodes	814	2649	13,115	21,270	0
Time (s)	0.615	0.861	2.799	2.719	0.394

4 Results

The tests were run on an Intel Core i7-2600—3.40 GHz computer using CPLEX solver Version 12.5. The model was implemented in the Optimization Programming Language (IBM ILOG OPL).

To build the instances we considered the number of orders $E = 10$, the time horizon $T = 10$ days, the number of silos $S = 5$. The capacity of each silo $CS_s = 3000$ kg. The capacities CT, CEG, CM, CEM_1 and CEM_2 were all fixed to 3000 and the yields and T_p were fixed to 1 for all products. To build the blends, matrix B_{pc}, we used the blends that the company produces, with $P = 24$ products made out of $C = 22$ types of coffee.

For the ten orders we have chosen ten different products and random finishings, the same for all the instances. The delivery dates changed for each instance and are distributed along the time horizon. All the instances have 100 real variables x_{et}, 220 real variables CA_{ct}, 1100 binary variables δ_{sct} and at most 560 constraints.

The quantities considered for the orders of each instance, quantity of order and the results obtained are presented in Table 1. For all these small, aggregate, examples the optimal solution was found in less than 3 s. This is an encouraging result as the company wants to embed this model in a DSS to accept new orders.

5 Conclusions and Future Work

The results obtained using the production planning model presented in this paper are quite interesting and aligned with the needs of the company. For the problem dimension that have to be solved daily in the company the running time is very low, which enables the daily use of the model embedded in a DSS to analyse the implications in the production plan of accepting additional orders (or proactively triggering them) whilst determining the silo loads and the production quantities.

As future work we intend to introduce in the model additional constraints on the batch production in the roaster. They were not taken into account in the model presented in this paper because the company did not consider the batches in the roaster a bottleneck of their production process.

Acknowledgements Project "NORTE-07-0124-FEDER-000057" is financed by the North Portugal Regional Operational Programme (ON.2 – O Novo Norte), under the National Strategic Reference Framework (NSRF), through the European Regional Development Fund (ERDF), and by national funds, through the Portuguese funding agency, Fundação para a Ciência e a Tecnologia (FCT).

References

1. Cooper, M.C., Lambert, D.M., Pagh, J.D.: Supply chain management: more than a new name for logistics. Int. J. Logist. Manag. **8**(1), 1–14 (1997)
2. Donnet, L.M.: Essays on specialty coffee procurement. Ph.D. thesis, Michigan State University (2007)
3. Govil, M., Proth, J.M.: Definition of a supply chain. In: Govil, M., Proth, J.M. (eds.) Supply Chain Design and Management, pp. 7–16. Academic, San Diego (2002)
4. Hicks, A.: Post-harvest processing and quality assurance for speciality organic coffee products. AU J. Technol. **5**(3), 22–26 (2002)
5. Lewin, B., Giovannucci, D., Varangis, P.: Coffee markets: new paradigms in global supply and demand. Discussion Paper 3, pp. 1–133. World Bank Agriculture and Rural Development, Washington, DC (2004)
6. Murthy, P.S., Madhava Naidu, M.: Sustainable management of coffee industry by-products and value addition. A review. Resour. Conserv. Recycl. **66**, 45–58 (2012)
7. Oestreich-Janzen, S.: Chemistry of coffee. In: Mande, L., Ben, H.W. (eds.) Comprehensive Natural Products II, chap. 3.25 - Che, pp. 1085–1096. Elsevier, Oxford (2010)
8. Ponte, S.: The 'latte revolution'? regulation, markets and consumption in the global coffee chain. World Dev. **30**(7), 1099–1122 (2002)
9. Taylor, P.: In the market but not of it: fair trade coffee and forest Stewardship council certification as market-based social change. World Dev. **33**(1), 129–147 (2005)
10. Thurber, F.B.: Coffee from Plantation to Cup. American Grocer Publishing Association, New York (1881)
11. Viere, T., Enden, J., Schaltegger, S.: Life cycle and supply chain information in environmental management accounting: a coffee case study. In: Burritt, R., Schaltegger, S., Bennett, M., Pohjola, T., Csutora, M. (eds.) Environmental Management Accounting and Supply Chain Management, pp. 23–40. Springer, Dortrecht (2011)
12. Vitasek, K.: Supply Chain Management: Terms and Glossary, p. 186. Council of Supply Chain Management Professionals (CSCMP), Illinois (2013)
13. Wong, H., Potter, A., Naim, M.: Evaluation of postponement in the soluble coffee supply chain: a case study. Int. J. Prod. Econ. **131**(1), 355–364 (2011)

Optimization of Production Scheduling in the Mould Making Industry

Bárbara Esperança Virgílio, Marta Castilho Gomes, and Ana Paula Barbosa-Póvoa

1 Introduction

The mould making industry plays a ubiquitous role in modern life, as the manufacture of many products of everyday use depends on moulds. Making these tools is a high-tech process with a well-established base in Portugal, one of the largest exporters of moulds in the world. In this industry the product (mould) is individually designed and produced in accordance with the customer's specifications (make-to-order, one-of-a-kind production). The time span for designing and producing a mould has shortened considerably as the lifecycles of products produced by moulding have also decreased. In order to improve competitiveness, many mould producers are now looking to develop solutions for improving the efficiency of their processes and optimizing time and costs (Ni et al. [8]). These solutions include the development of computational tools to support production planning and scheduling.

In contrast to other discrete production industries, such as electronic chip or semiconductor production, for which the scientific literature dealing with the respective production planning and scheduling is abundant, research specific to the mould making industry is very limited. As Choy et al. [1] point out, production scheduling in this industry is still normally carried out using traditional methods.

B.E. Virgílio
GECO, Rua da Arroteia, Telheiro, 2405-032 Maceira, Portugal
e-mail: esperanca.barbara@gmail.com

M.C. Gomes (✉)
CESUR, Instituto Superior Técnico, Universidade de Lisboa, 1049-001 Lisboa, Portugal
e-mail: marta.gomes@tecnico.ulisboa.pt

A.P. Barbosa-Póvoa
CEG-IST, Instituto Superior Técnico, Universidade de Lisboa, 1049-001 Lisboa, Portugal
e-mail: apovoa@tecnico.ulisboa.pt

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_21

The main objectives of this research were to develop an integer linear programming model to support production scheduling in the mould industry and solve it using real data supplied by a company (GECO). The decisions taken into account in the model regard assignment of operations to the shop floor machines and scheduling the production of each mould part, whilst meeting the due dates agreed up with the customers.

2 Context of Motivating Case Study

GECO is a firm specialized in injection moulds for the plastics industry that began operating in 1969. It is currently part of an international industrial group with headquarters in the district of Leiria, Portugal. The company employs over 300 people and currently exports almost all of its output, which goes mostly to the automobile (80 % of total production), electronics, packaging, pharmaceutical and home appliances industries. The turnover was EUR 14 million in 2013.

At GECO, production planning and scheduling is the responsibility of the production manager, the appointed Group Leader (GL). This work is carried out manually, using Excel worksheets and the Microsoft Project software for the storage and visualization of data. The scheduling decisions are taken with no help of a computer implemented algorithm. Based on this, the challenge was to develop a model that could address in an efficient way the production scheduling at the company.

3 OPTMESC Model Characterization

The mould making production scheduling problem pertains to the job shop category, a production environment where the range of products is diverse, production numbers are small and orders may follow different production routes. Jain and Meeran [7] and, more recently, Potts and Strusevich [10] present reviews of solution methods for the job shop scheduling problem proposed in the literature. The classical job shop scheduling problem, in which there is only one machine per operation, has been the object of intensive study. However, as Tay and Ho [11] point out, on the shop floor there are in practice multiple copies of the most critical machines, so as to minimize bottleneck situations resulting from time-consuming operations or busy machines. This generalization is named the flexible job shop scheduling problem.

An analysis of that literature shows that, whilst the majority of authors propose mathematical programming formulations for job shop scheduling, they apply heuristic or meta-heuristic techniques in solving the problem. The interest in solution methods based on mathematical programming is quite recent. Özgüven et al. [9], Gomes et al. [6] and Demir and Işleyen [2] are recent examples in the literature of mathematical programming models application to the flexible job shop scheduling problem.

OPTMESC model is based on the discrete time model for flexible job shop proposed in Gomes et al. [4] and Gomes [3] , which has been adapted to take the characteristics of the case study into account. The production scheduling problem and how it is modelled by OPTMESC can be described as follows (note that the Appendix further extends the problem description):

- A mould consists of various parts, which differ from mould to mould. Each part follows a production route, and different parts may follow the same production route.
- There may be some exactly identical parts, i.e. two or more copies of the same part in a mould. In such a case, the model considers them to be two or more units of the same part (they are modelled together in terms of decision variables).
- A production route is made up of several operations in sequence. Although there is a generic operations flow, not all operations may be involved in the production route for a specific part. Figure 2 in the Appendix depicts the production routes for an illustrative example of a mould.
- Each operation is performed by a machine and there are different machines available for performing a given operation (machines in parallel). In addition, a machine may perform different operations.
- Each operation is preceded by a buffer or waiting queue in which the parts await processing. The buffer is common to all machines that can perform the operation in question.
- The machines are characterized by the operation or operations they carry out, the processing times and capacity (total number of part units they can process simultaneously). Processing times depend on the part, operation and machine, and take integer values.
- Each mould has a due date. Production of the mould parts should be scheduled to meet the due date for the mould.

All decision variables are integer. The scheduling horizon is divided into identical time slots leading to a discrete time model. The objective function, to be minimized, is a sum of penalties: penalties for incomplete parts at the end of the scheduling horizon; penalties for finishing production of parts before or after the mould due date and penalties for waiting times (of unfinished parts) in intermediate buffers. The set of constraints comprises[1]:

- Starting constraints which define conditions of machines and buffers at the start of the scheduling horizon.
- Flow balance equations that establish the relationship between part units in buffers in adjacent time intervals. They have different forms for the first and final

[1]The model formulation is presented in [12] and [13].

buffer in a processing route while the constraints for all the intermediate buffers in a route have the same form (see the Appendix).

- Machine capacity constraints ensure, for all time slots of the scheduling horizon, that the total number of part units loaded into a machine does not exceed its capacity.
- Buffer capacity constraints, also written for every slot of the scheduling horizon, limit the total number of part units lodging in the buffer of an operation to the available buffer capacity.

4 Application to a Production Plan

Based on the motivating case study, OPTMESC model was applied to data from a GL production plan and the model solution compared with it. To this end, data on the characteristics of the GECO machinery fleet were collected (i.e., a list of the operations that can be performed on each machine and the maximum dimensions of the parts it can take); and the operations and respective production times were gathered for all parts involved in the GL production plan. The plan consists of 42 parts belonging to six moulds and involves assignment of operations to nine machines (all with capacity for a single part unit). Operations considered are grinding/rough milling, DNC machining of mould structures, DNC machining of moulding parts, drilling for water run-off and drilling for mould assembly and/or extraction. For implementation in the OPTMESC model, a correspondence between machines and operations was established that took into account the size of parts that can be loaded into the machines. Twelve production routes were defined for the parts, involving a total of 52 operations. The complete set of production routes and the respective times are described in [12]. In deciding the production plan, the GL relies on his own experience for defining the operations (which vary according to the type of part) and for estimating the respective production times.[2] Buffer capacity was defined at 100 units (a high figure because the scheduling at GECO does not take into account space constraints).

The parts and due dates are shown in Table 1, where each line corresponds to a mould. One unit of each part is to be produced and the penalties in the objective function are based on [3] : 20 for completion after the due date; 1 for completion before the due date; 0.1 for waiting time in intermediate buffers and 10^7 for unfinished parts at the end of the scheduling horizon. The latter coefficient is very high in order to avoid the unfinished part scenario. The GL was asked as to the recommended values for these penalties but there was no counterproposal from him.

[2]The GL practical knowledge of the various stages in the process is needed for this "pre-processing" of the operations for determining the production plan.

Table 1 Data of parts for six moulds (plan A1 + additional parts in plan B1)

Parts	Due date
p1, p2, p3, p4, p5, p6, p7, p8	26
p9, p10, p11, p12, p13, p14, p15	80
p16, p17, p18, p19, p20, p21 + (p43, p44, p45, p47—plan B1)	76
p22, p23, p24, p25, p26, p27, p28, p29	86
p30, p31, p32, p33, p34, p35, p36 + (p49—plan B1)	102
p37, p38, p39, p40, p41, p42 + (p46, p48—plan B1)	102

Table 2 Numerical characteristics of the models and results for production plans A1 and B1

Plan	No. of parts	No. of variables	No. of constraints	No. of iterations	No. of nodes	Objective function	CPU time (s)	Time in buffers	Time before date	Time after date
A1	42	30,401	18,164	4992	786	384.60	4.62	186	366	0
B1	49	34,302	20,490	16,983	6	485.40	3.29	234	462	0

The GL production plan corresponds to a time frame of 51 days, i.e., 2 months and 9 days of manufacturing (not counting weekends), where the production times are expressed as multiples of 6 h (one half of a work day, which has 12 h). Accordingly, in the time grid used in the OPTMESC model, the time slot equals one half-day, and so the due dates in Table 1 are expressed in half-days.

In the GL plan, all machines except for 2 were unavailable at the start of the scheduling horizon. Two distinct situations were studied in determining the production plan, named plans A1 and B1. The former takes into account only those parts in the GL plan produced internally at GECO. For the latter, parts in the GL plan that were outsourced were also included. Indeed, the GL production plan contained 56 parts, of which 42 were produced in-house and 14 were outsourced because the capacity of the machines was thought insufficient for meeting the due dates.

The scheduling horizon is 105 half-days (slightly longer than the latest due date in Table 1). In plan B1 it was feasible to produce 7 of the 14 outsourced parts (49 parts in total). This required the addition of new production routes for five parts; the other 2 used production routes already defined. OPTMESC model was implemented in GAMS modelling system and solved with CPLEX 12.3 release, on a 2.20 GHz Intel Core with 6 GB of RAM running Windows 7. Table 2 presents the numerical characteristics of the models and the results (for optimal solutions), which include the objective function components before multiplication by the penalties.

OPTMESC was solved to optimality in less than 5 s in both cases with neither incomplete parts at the end of the scheduling horizon nor parts finished after the due date. The objective function value is higher for B1 plan because of the increase in the total waiting time for parts in the buffers and parts produced before the due date. The GL takes about 1 h to plan production of a similar set of parts and hence the use of OPTMESC model could lead to significant time savings in defining production plans at GECO.

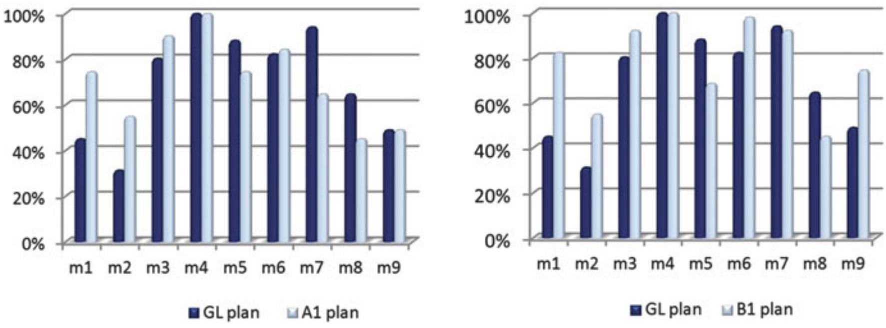


Fig. 1 Machine loads in GL, A1 and B1 plans

Figure 1 shows a comparison between the GL plan and plans A1 and B1 regarding machine load. The average machine load in the GL and A1 plans is 71 %, however the standard deviation is 24 % in the former and 19 % in the latter. Therefore, OPTMESC produced a more uniform solution regarding the assignment of operations to the machines. Due to the insertion of additional parts, B1 plan presents a higher average machine load (79 %) but the standard deviation (20 %) is still very close to that observed for plan A1.

5 Post-optimization Analysis

A post-optimization analysis of the objective function coefficients, the due dates of orders (moulds) and the length of the scheduling horizon was carried out in Virgílio et al. [13], since these are the OPTMESC model parameters with the most relevant effect on the objective function value and computation time.

Results for the order due date study are summarized in Table 3. Based on the B1 solution (B1 plan), the impact on the optimal solution of compressing the due dates was assessed. The due date for the first mould (26 half-days, first line in Table 1) was not changed but the other due dates were shortened simultaneously by 10, 20, 30 and 40 half-days. The scheduling horizon of 105 half-days and the objective function penalties of solution B1 were kept when solving the model.

Table 3 shows that when the due dates are compressed the objective function worsens (increases) and the CPU time needed to solve the model to optimality increases. The exception is the transition between solutions B1 and B2, where the CPU time for the latter is slightly shorter. The last three columns show an increase in the total time of production after the due date (and hence of the objective function), although the total time in buffers and total time before the due date were actually reduced. Solutions B1 to B4 present CPU times that do not exceed 40 s, whereas for solution B5 this is 656 s (about 11 min), and the number of nodes analysed in the search tree reached 19,123, a significantly higher figure than the ones of the previous solutions.

Table 3 Results of the order due date analysis

Solution	Change in dates	No. of variables	No. of con-straints	No. of itera-tions	No. of nodes	CPU time (s)	Objective function	Time in buffers	Time before date	Time after date
B1	0			16,983	6	3.32	485.40	234	462	0
B2	−10			17,433	0	2.01	566.20	222	504	2
B3	−20	34,302	20,490	51,012	110	10.51	1832.20	82	464	68
B4	−30			244,216	2148	39.38	3980.60	56	375	180
B5	−40			2,028,778	19,123	655.59	7193.00	30	290	345

Note: These figures are valid for all solutions (B_1 to B_5) and not only for solution B_3, as the table seems to convey.

By progressively reducing the scheduling horizon length, starting from solution B5:

- For scheduling horizons between 104 and 94 half-days the optimal solutions present the same objective function value and objective function components (total time in buffers and time before and after the due date) as solution B5.
- For 93 half-days, the objective function value increased to 7745.40 and the optimal solution was computed in a considerably shorter CPU time (137 s versus 656 s).
- For 92 half-days, there was one unfinished part at the end of the scheduling horizon and so there was a steep increase in the optimum objective function value (to 10,006,262.30), due to the penalty of unfinished parts being 10^7. The model was significantly more difficult to solve: the CPU time rose to 1272 s (around 21 min), with the number of nodes analysed being almost two times the one of solution B5.
- However, if we allow for sub-optimality in this case, good solutions can be obtained in significantly lower CPU time. Indeed, for optimality gaps of 5 or 10 % the solver finds a solution in 304 s (about 5 min), both solutions having the same objective function value and objective function components.

6 Conclusion and Future Work

This study proposes an integer linear programming model, OPTMESC, for solving the problem of production scheduling in the mould making industry. This discrete time model was applied to real data provided by a mould making company to determine a production plan, which involved the production of various parts of different moulds. Using GAMS modelling system and the solver CPLEX, optimal solutions were computed for two instances of the problem in a matter of seconds. A comparison with the production manager's plan showed that, based on the available machine capacities, it is possible to schedule production of a significant number of additional parts that were outsourced in that plan. This is an important result, as reducing the level of outsourcing of parts reduces the level of uncertainty as to their quality and the compliance with the due dates.

The post-optimization analysis carried out illustrates, on the one hand, the increase in computational complexity (and so, in the CPU time needed to solve the model) when the due dates and the scheduling horizon are compressed. On the other hand, it also shows the tests a decision-maker can easily do with OPTMESC, such as the impact of changing the mould due dates (for instance to adapt to the available machine capacity) and of considering different time horizons in the production schedule. The decision-maker can then choose one of the several solutions generated by OPTMESC.

The model thus proved adequate to be included in a decision support system (DSS) for scheduling in the mould-making industry, where the problem is still commonly solved by manual methods. Implementation of such DSS tool requires the creation of a user-friendly, interactive and flexible interface that enables direct visualization of the machine loading diagram (Gantt chart) based on the model results. Conceiving such an interface is one direction of work in the future.

Another direction is generalizing the OPTMESC model. Firstly, the generalization that models the decision as to what parts to outsource in the event of lack of in-house capacity should be explored (a simple heuristic was used here for selecting the parts in plan B1). A second aspect is the extension to the re-scheduling problem (re-scheduling to take changes into account), which was studied in [3] and [5] for a discrete time model. This includes the need for mould corrections (very common in this industry) and the insertion of new orders (moulds) into an existing plan, making use of the available machine capacity.

Appendix: Extension of the Problem Description in Section 3

- The main parts of a mould are the cavity (concave part) and the punch or core (convex part). The material to be moulded is placed in a pre-heated state in the space between them and hardens, getting the desired shape.
- A production route is made up of several operations in sequence. Although there is a generic operations flow, not all operations may be involved in the production route for a specific part.
- Figure 2 presents the production routes for an illustrative example of a mould. Operations considered are grinding or rough milling (Grind), DNC[3] machining of the mould structure (DNC_S), DNC machining of moulding parts (DNC_M), rectification grinding (Rectif), electro-erosion (Erosion), drilling with the purpose of water run-off (Drill (wat)) and drilling with the purpose of mould assembly and/or extraction (Drill (ext)).
- Each operation is performed by a machine. There are different machines available for performing a given operation. In addition to this, a machine may perform different operations. For example, a machine that does the rough milling may also do the DNC machining of the mould structure.
- OPTMESC model uses the notions of initial buffer (the one for the first operation), intermediate buffers (for the remaining operations) and final buffer (for finished parts) in each production route. Figure 2 depicts the alternating sequences of buffers and operations in the two production routes of the example.

[3]Direct numerical control.

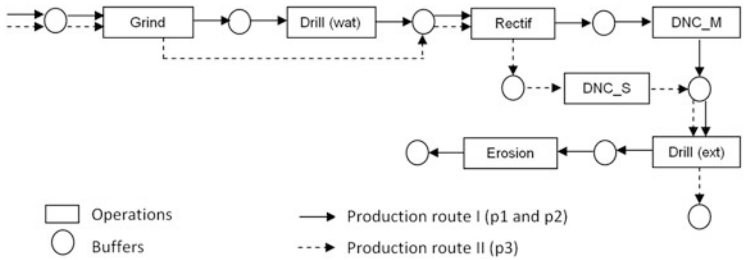


Fig. 2 Production routes for an illustrative, realistic example of a mould comprising three parts (p1, p2 and p3)

References

1. Choy, K.L., Leung, Y.K., Chow, H.K.H., Poon, T.C., Kwong, C.K., Ho, G.T.S., Kwok, S.K.: A hybrid scheduling decision support model for minimizing job tardiness in a make-to-order based mold manufacturing environment. Expert Syst. Appl. **38**, 1931–1941 (2011)
2. Demir, Y., Işleyen, S.K.: Evaluation of mathematical models for flexible job-shop scheduling problems. Appl. Math. Model. **37**, 977–988 (2013)
3. Gomes, M.C.: Reactive scheduling in make-to-order production systems: an optimization based approach. Ph.D. thesis, Instituto Superior Técnico, Universidade Técnica de Lisboa (2007)
4. Gomes, M.C., Barbosa-Póvoa, A., Novais, A.Q.: Optimal scheduling for flexible job shop operation. Int. J. Prod. Res. **43**, 2323–2353 (2005)
5. Gomes, M.C., Barbosa-Póvoa, A., Novais, A.Q.: A discrete time reactive scheduling model for new order insertion in job shop, make-to-order industries. Int. J. Prod. Res. **48**, 7395–7422 (2010)
6. Gomes, M.C., Barbosa-Póvoa, A., Novais, A.Q.: Reactive scheduling in a make-to-order flexible job shop with re-entrant process and assembly: a mathematical programming approach. Int. J. Prod. Res. **51**, 5120–5141 (2013)
7. Jain, A.S., Meeran, S.: Deterministic job shop scheduling: past, present, future. Eur. J. Oper. Res. **113**, 390–434 (1999)
8. Ni, Q., Feng, L.W., Yarlagadda, P.K., Ming, X.: Business information modeling for process integration in the mold making industry. Robot. Comput. Integr. Manuf. **23**, 195–207 (2007)
9. Özgüven, C., Yavuz, Y., Özbakır, L.: Mixed integer goal programming models for the flexible job-shop scheduling problems with separable and non-separable sequence dependent setup times. Appl. Math. Model. **36**, 846–858 (2012)
10. Potts, C., Strusevich, V.: Fifty years of scheduling: a survey of milestones. J. Oper. Res. Soc. **60**, S41–S68 (2009)
11. Tay, J.C., Ho, N.B.: Evolving dispatching rules using genetic programming for solving multi-objective flexible job-shop problems. Comput. Ind. Eng. **54**, 453–473 (2008)
12. Virgílio, B.E.: Optimização do planeamento e escalonamento da produção na indústria de moldes: Aplicação ao caso da geco. MSc dissertation in Management and Industrial Engineering, Instituto Superior Técnico, Universidade Técnica de Lisboa (2011).
13. Virgílio, B.E., Gomes, M.C., Barbosa-Póvoa, A.: Optimização do escalonamento da produção na indústria de moldes: Aplicação ao caso da geco. In: Oliveira, R.C., Ferreira, J.S. (eds.) Investigação Operacional em Ação — Casos de Aplicação, pp. 199–245. Imprensa da Universidade de Coimbra, Coimbra (2014)

Constraint Aggregation in Non-linear Programming Models for Nesting Problems

Pedro Rocha, A. Miguel Gomes, Rui Rodrigues, Franklina M.B. Toledo, and Marina Andretta

1 Introduction

The Nesting problem is a complex problem that arises in industries where sets of pieces or space must be efficiently placed or allocated in order to minimize wasted space or wasted raw materials, without overlaps between pieces and fully contained inside a container. It is a 2D Cutting and Packing problem where pieces have non-regular geometries, also known as the Irregular Strip Packing problem.

The Nesting problem has a geometrical and combinatorial component, where the first impacts the approaches required to tackle the second. The selection of an adequate geometrical representation is an important issue to reduce the complexity of the geometric component. The geometric representation should be able to deal with continuous rotations and allow easy overlap computations.

Several geometrical representations are discussed in [3], each one tailored to a specific application: grid representation for discrete position placement and orthogonal orientations; polygonal representation (including No-Fit-Polygons) use polygons for continuous position placement and arbitrary discrete orientations; and Φ-Functions [9] for continuous position and orientation placement of the pieces, using sets of primary pieces composed by straight lines and arc segments. An alternative representation is to approximate irregular pieces by a Circle Covering representation. In [4] a grid is used to place the circles, a three-step algorithm to

P. Rocha • A.M. Gomes (✉) • R. Rodrigues
INESC-TEC, Faculdade de Engenharia, Universidade do Porto, Porto, Portugal
e-mail: pedro.f.rocha@inesctec.pt; agomes@fe.up.pt; rui.rodrigues@fe.up.pt

F.M.B. Toledo • M. Andretta
Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Paulo, Brazil
e-mail: fran@icmc.usp.br; andretta@icmc.usp.br

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_22

approximates pieces by circles is proposed in [11], a greedy heuristic to place the circles is used in [6], and a circle covering algorithm based on Medial Axis is used in [7] to minimize the number of circles needed to represent a piece.

Several works use Non-Linear Programming models (NLP) together with circle covering representations of pieces. In [5] an Iterated Local Search algorithm is proposed where a NLP model is used to remove overlap between pieces. In [10] the authors propose a NLP mathematical model for strip packing supporting circles and non-convex polygons (used in conjunction with Φ-Functions). Global optimization methods based on quadratic mathematical models and able to optimally solve small instances (4–6 pieces) are proposed in [6]. In [8] the usage of a NLP model and several circle covering representations to solve medium size Nesting problems with continuous rotations (up to 50 pieces) is explored.

The use of NLP models in conjunction with a Circle Covering representation allows tackling the Nesting problem with continuous rotations. The main difficulty lies when dealing with large size instances, due to the large number of pieces which cause an exponential growth in the non-overlapping constraints derived from the comparisons between pairs of pieces. In this work we propose a method that aggregates non-overlapping constraints to reduce the computational cost without significant impact on the quality of the layout solutions, which allows NLP models to tackle larger instances. The proposed approach uses a complete circle covering representation presented in [7] and introduces a pieces based NLP model, derived from the model presented in [8].

2 Non-linear Programming Model

A mathematical model based on a circle covering representation for the Nesting problem with continuous rotations requires NLP models due to the non-linearity of the distance computation and the trigonometric operations. In the proposed model, each piece is represented by a set of circles with fixed relative positions between each other. Three variables are needed for each piece, two for the placement position on the layout and one for the orientation. The model has a non-overlapping constraint for each pair of circles from different pieces and four Containment constraints for each circle, one for each side of the container. The mathematical formulation of the model is the following:

$$\text{minimize} \quad l \tag{1}$$

$$\begin{aligned}\text{subject to:} \quad & (R_{k_i} + R_{h_j})^2 - (x_k + \cos(A_{k_{0,i}} + \theta_k) \times D_{k_{0,i}} - \\ & - x_h - \cos(A_{h_{0,j}} + \theta_h) \times D_{h_{0,j}})^2 - \\ & - (y_k + \sin(A_{k_{0,i}} + \theta_k) \times D_{k_{0,i}} - \\ & - y_h - \sin(A_{h_{0,j}} + \theta_h) \times D_{h_{0,j}})^2 \leq 0,\end{aligned}$$

$$\forall i \in C_k,\ \forall j \in C_h, \forall k, h \in \mathbb{N}, k \neq h \qquad (2)$$

$$x_k + \cos(A_{k_{0,i}} + \theta_k) \times D_{k_{0,i}} + R_k - l \leq 0, \qquad \forall i \in C_k,\ \forall k \in \mathbb{N} \qquad (3)$$

$$R_k - x_k - \cos(A_{k_{0,i}} + \theta_k) \times D_{k_{0,i}} \leq 0, \qquad \forall i \in C_k,\ \forall k \in \mathbb{N} \qquad (4)$$

$$y_k + \sin(A_{k_{0,i}} + \theta_k) \times D_{k_{0,i}} + R_k - W \leq 0, \qquad \forall i \in C_k,\ \forall k \in \mathbb{N} \qquad (5)$$

$$R_k - y_k - \sin(A_{k_{0,i}} + \theta_k) \times D_{k_{0,i}} \leq 0, \qquad \forall i \in C_k,\ \forall k \in \mathbb{N} \qquad (6)$$

$$x_k, y_k, \theta_k, l \in R \qquad (7)$$

The Objective Function (1) aims to minimize the length of the strip and is represented by the auxiliary variable l. Each piece k is composed by a set of C_k circles and is represented by three variables: x_k and y_k defines the position of piece k on the layout and θ_k defines its orientation. The number of variables grow linearly with the number of pieces. The non-overlapping constraint (2) compare the distance between each pair of circles i, j from pieces k, h and the sum of radius of both circles R_{k_i} and R_{h_j}. The containment constraints (3)–(6) ensure that each circle does not exceed the admissible placement region. The number of non-overlapping constraints has a factorial growth, while the number of containment constraints grows linearly. Finally, the variable domains are defined in (7).

3 Aggregating Non-overlapping Constraints

The NLP model presented in the previous section has difficulties in solving large instances due to high computational cost caused by the large number of non-overlapping constraints, which is caused by the increase in the number of pieces and in the number of circles necessary to represent them. To overcome this difficulty we propose to aggregate all non-overlapping constraints in a single summation constraint, where each summation term derives from constraint (2).

The aggregation of non-overlapping constraints allows discarding a large number of summation terms between distant pairs of pieces, by considering their terms zero (8). This is possible because the original non-overlapping constraint (2) will return a negative value. This also has the additional benefit of reducing the internal computational cost of the solver, when compared to the original model with independent non-overlapping constraints. Each summation term represents the difference between the squared sum of the radius of both circles R_{k_i} and R_{h_j} (9) and the squared distance between each pair of circles i, j from pieces k, h (10)–(11).

$$\sum_{i=0}^{C_k}\sum_{j=0}^{C_h}\sum_{k=0}^{N}\sum_{h=0}^{N}\left\{\max\left[0, \mathrm{NOVLPR}_{i,j,k,h} - (\mathrm{NOVLPX}_{i,j,k,h} + \mathrm{NOVLPY}_{i,j,k,h})\right]\right\}^2 \leq 0, \tag{8}$$

$$\forall i \in C_k,\ \forall j \in C_h,\ \forall k, h \in \mathbb{N},\ k \neq h,\ x_{k_i}, y_{k_i},\ \theta_k, l \in R$$

$$\mathrm{NOVLPR}_{i,j,k,h} = (R_{k_i} + R_{h_j})^2 \tag{9}$$

$$\mathrm{NOVLPX}_{i,j,k,h} = (x_{k_i} + \cos(A_{k_{0,i}} + \theta_k) \times D_{k_{0,i}} - x_{h_j} - \cos(A_{h_{0,j}} + \theta_h) \times D_{h_{0,j}})^2 \tag{10}$$

$$\mathrm{NOVLPY}_{i,j,k,h} = (y_{k_i} + \sin(A_{k_{0,i}} + \theta_k) \times D_{k_{0,i}} - y_{h_j} - \sin(A_{h_{0,j}} + \theta_h) \times D_{h_{0,j}})^2 \tag{11}$$

Summation terms are discarded by using spatial partition and hierarchical overlapping detection. Spatial partition divides the space into regions, and only compares pieces from the same or adjacent regions. The hierarchical overlap detection uses a three level representation, where the first level uses an orthogonal bounding box, the second level uses a minimum enclosing circle (MEC), and the last uses the circle covering. Comparisons start by using the basic piece representations and when it cannot discard overlap, it checks overlap with higher quality representations.

4 Results and Discussion

To evaluate the aggregation of non-overlapping constraints a set of five nesting instances were selected from the ESICUP website (http://www.fe.up.pt/esicup), especially suited for continuous rotations. Instance *poly1a* is the least complex one with only 15 pieces. Instances *poly2a* and *poly3a* are multiples of *poly1a* with respectively 30 and 45 pieces (i.e., 2 and 3 times the pieces of *poly1a*). Instances *poly2b* and *poly3b* have respectively 30 and 45 pieces, but only 15 are the pieces of *poly1a* and the remaining ones are new pieces. The circle coverings were obtained by the algorithm proposed in [7].

The computational experiments were performed on a computer with two Intel Xeon E5-5690 processors at 3.46 GHz, with 48 Gb Ram at 1333 MHz, running Ubuntu 12.04 LTS x86-64, and using a single-thread. The selected Non-linear solver is Algencan v2.37 [1, 2] (http://www.ime.usp.br/~egbirgin/tango/), which is a non-linear solver based on the Augmented Lagrangian multipliers method.

The selected solver converges to a local minimum, requiring multiple starting points to explore different regions of the solution space. A total of 30 initial solutions were generated by randomly placing the pieces on a grid, in a non-overlapping configuration with random rotations.

Table 1 Model variants results

Instance	Number of pieces	Model variant	Obj. Function (l)			Avg. time (s)
			Min.	Avg.	Max.	
poly1a	15	*independent*	16.39	17.33	18.57	24.47
		aggregated	16.04	17.63	19.09	9.00
poly2a	30	*independent*	30.30	32.25	35.26	635.70
		aggregated	30.94	32.84	34.58	111.10
poly2b	30	*independent*	33.83	35.64	38.05	709.17
		aggregated	33.64	35.78	38.51	121.40
poly3a	45	*independent*	45.70	47.33	48.99	3104.67
		aggregated	46.67	47.96	49.65	249.57
poly3b	45	*independent*	45.00	47.02	50.01	2736.03
		aggregated	46.37	48.03	55.86	266.47

Two model variants were tested, one with the model (1)–(7) where the non-overlapping constraints are considered independent, and the other where the non-overlapping constraints are aggregated in a single constraint (8), respectively denoted as *independent* and *aggregated*. Table 1 summarizes the results obtained, where the first column identifies the instance, the second one shows the number of pieces for each instance, and the third column identifies the model variant. The next two columns presents the minimum and average objective function (layout length) achieved by each model variant for the 30 initial solutions. The last column shows the average running time, in seconds, of each variant.

The results show that both variants have distinct behaviours in what concerns solution quality and computational time. Aggregating non-overlapping constraints clearly allows the computational times to be much lower, allowing this variant to scale well for instances with more pieces. The average computational time of the aggregated variant is less than 10 % of the variant with the original model for the larger instances. This is achieved at expenses of solution quality which in average decreases for the aggregated variant in all instances (the biggest decrease is 2.1 % for *poly3b*). The minimum layout length shows a different behaviour, with the aggregated variant achieving better results for *poly1a* and *poly2b*, and worst results for the remaining instances (the biggest decrease is 3.0 % for *poly3b*).

5 Conclusions

The main conclusion of this work is that aggregating non-overlapping constrains proved to be an effective and efficient technique to handle the computational complexity of larger instances. The downside of this approach is a small reduction in the layout quality. If the objective is to obtain the best layout length, the best option to use is the variant with independent non-overlapping constraints. However, if the aim is to solve instances with a large number of pieces in a reasonable computational time, the best choice is to use the model variant with aggregated non-overlapping constraints at expenses of a slight decrease in the solution quality.

Acknowledgements This research was partially supported by Fundação para a Ciência e Tecnologia (FCT) through Project PTDC/EME-GIN/105163/2008 (EaGLeNest) and ERDF through the Programme COMPETE (COMP-01-0124-FEDER-010176).

References

1. Andreani, R., Birgin, E., Martínez, J., Schuverdt, M.: On augmented lagrangian methods with general lower-level constraints. SIAM J. Optim. **18**, 1286–1309 (2007)
2. Andreani, R., Birgin, E., Martínez, J., Schuverdt, M.: Augmented lagrangian methods under the constant positive linear dependence constraint qualification. Math. Program. **111**, 5–32 (2008)
3. Bennell, J., Oliveira, J.: A tutorial in irregular shape packing problems. J. Oper. Res. Soc. **60**, 93–105 (2009)
4. Imamichi, T., Nagamochi, H.: A multi-sphere scheme for 2d and 3d packing problems. In: Engineering Stochastic Local Search Algorithms. Designing, Implementing and Analyzing Effective Heuristics, vol. 4638, pp. 207–211. Springer, Heidelberg (2007)
5. Imamichi, T., Yagiura, M., Nagamochi, H.: An iterated local search algorithm based on nonlinear programming for the irregular strip packing problem. Discret. Optim. **6**(4), 345–361 (2009)
6. Jones, D.R.: A fully general, exact algorithm for nesting irregular shapes. J. Glob. Optim. **59**, 1–38 (2013)
7. Rocha, P., Gomes, A., Rodrigues, R., Toledo, F.: Circle covering using medial axis. In: Proceedings of the 11th IFAC Workshop on Intelligent Manufacturing Systems, vol. 11, pp. 402–407 (2013)
8. Rocha, P., Gomes, A., Rodrigues, R., Toledo, F., Andretta, M.: Circle covering representation for nesting problems with continuous rotations. In: Proceedings of the 19th IFAC World Congress on International Federation of Automatic Control (IFAC) 2014, vol. 19, pp. 5235–5240 (2014)
9. Stoyan, Y., Scheithauer, G., Gil, N., Romanova, T.: Phi functions for complex 2d-objects. Q. J. Belg. Fr. Ital. Oper. Res. Soc. **2**(1), 69–84 (2004)
10. Stoyan, Y., Zlotnik, M., Chugay, A.: Solving an optimization packing problem of circles and non-convex polygons with rotations into a multiply connected region. J. Oper. Res. Soc. **63**(3), 379–391 (2012)
11. Zhang, W., Zhang, Q.: Finite-circle method for component approximation and packing design optimization. Eng. Optim. **41**, 971–987 (2009)

Part IV
Optimization Methods

A Hybrid Genetic Algorithm for the One-Dimensional Minimax Bin-Packing Problem with Assignment Constraints

Mariona Vilà and Jordi Pereira

1 Introduction and Problem Definition

Bin-packing problems generally consist in the assignment of items to bins in a way that optimises a measure of efficiency. The one-dimensional minimax bin-packing problem with assignment constraints can be defined as follows: suppose T sets ($1 \leq t \leq T$) each one of them containing B ($1 \leq r \leq B$) items with an associated weight, w_{rt}. The objective of the problem is to split all of the items of the different sets into B groups or bins ($1 \leq b \leq B$) so that every group contains exactly one item of each set, while minimising the maximum sum of the weights of the items in any group.

Although classical bin-packing problems have been extensively studied in the past, see [4] or [5] for some classical works, the one-dimensional minimax bin-packing problem with assignment constraints has only recently been introduced, see [1]. Despite its recent history, the problem has significant applications in the area of psychology, especifically in the context of test design [7]. An example is the test-splitting problem. In this problem, the objective is to assign test questions (items), which are initially grouped into different sets, into several questionnaires (bins). In any solution, every questionnaire contains one question from each one of the original sets. Questions have an associated weight usually related to their difficulty,

M. Vilà (✉)
Escola Universitària d'Enginyeria Tècnica Industrial, Universitat Politècnica de Catalunya, C. Comte Urgell, 187 1 st. Floor, 08036 Barcelona, Spain
e-mail: mariona.vila.bonilla@upc.edu

J. Pereira
Departamento de Ingeniería Industrial, Universidad Católica del Norte, Av. Angamos 0610, Antofagasta, Chile
e-mail: jorgepereira@ucn.cl

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_23

and the optimal solution is the one which minimizes the maximum weight of any questionnaire.

This problem can be described by using the formulation recently proposed in [1] (see Eqs. (1)–(5)), which uses binary decision variables x_{rtb} that take value 1 when item r in set t is assigned to group b and 0 otherwise.

$$\text{minimize } Z \tag{1}$$

subject to:

$$Z \geq \sum_{r=1}^{B} \sum_{t=1}^{T} w_{rt} \cdot x_{rtb}, \text{ for } 1 \leq b \leq B; \tag{2}$$

$$\sum_{r=1}^{B} x_{rtb} = 1, \text{ for } 1 \leq b \leq B, \ 1 \leq t \leq T \tag{3}$$

$$\sum_{b=1}^{B} x_{rtb} = 1, \text{ for } 1 \leq r \leq B, \ 1 \leq t \leq T \tag{4}$$

$$x_{rtb} \in \{0, 1\}, \text{ for } 1 \leq r \leq B, \ 1 \leq b \leq B, \ 1 \leq t \leq T \tag{5}$$

Note that the numbering of the questionnaires in a solution is of no consequence. As such, the items of the first set can be prefixed (for example by setting x_{i1i} to 1 for all $1 \leq i \leq B$). This prefixing removes symmetries and reduces the number of variables of the formulation, as only the variables of the sets $[2, T]$ are needed.

This problem is NP-Hard, as the Odd-Even Partition problem and the 3-Partition problem can be reduced to this problem. Therefore, metaheuristic procedures seem a suitable choice for its resolution, as they are able to provide with good solutions in short computation times. The only metaheuristic proposed for the problem is a Simulated Annealing (SA) procedure, see [1]. In this paper, we propose a hybrid Genetic Algorithm (GA), which combines the principles of GA with the mathematical formulation proposed above, and we compare its performance with the previous approach.

The rest of this paper is structured as follows: Section 2 puts forward the characteristics of the GA proposed for the problem; and Sect. 3 presents the computational experiments performed on the implementation of the GA.

2 Description of the Genetic Algorithm

GAs [3] are metaheuristics inspired by natural selection, commonly used for solving optimisation problems. GAs usually explore the solution-space of a problem by using a set of solutions (individuals) known as the population. These individuals are used to create new individuals, which substitute the previous ones. To decide which individuals should be used to generate a new individual, and which of them should be substituted, the algorithm uses a fitness value related to the quality of the solution represented by that individual.

The following subsections give details about each characteristic of the steady-state, see [2], GA presented in this work.

2.1 Encoding and Fitness Function

The GA devised in this work uses a direct encoding in which each individual is represented by a two dimensional matrix with T columns and B rows. Each entry of the matrix indicates the questionnaire into which an item of a set is assigned. Two sample individuals for a problem with five sets and three questions per set are depicted in Fig. 1. In addition to the encoding, each entry also shows the weight of the item in parenthesis. Questionnaires are identified using a capital letter. In the figure, questionnaire A of the individual depicted on the left is composed of the first item of set 1, the second item of sets 2 and 3 and the third item of sets 4 and 5, for a total weight of 29.

The fitness of each individual corresponds to the inverse of the objective function. In the first example given in Fig. 1, the accumulated weights are 29 for questionnaire A, 32 for questionnaire B, and 19 for questionnaire C. The accumulated weights of the second parent are 23, 32 and 25 for questionnaires A, B and C, respectively. Therefore, the fitness value of both individuals is $1/32$.

2.2 Initialisation

In order to accelerate the search, the initial population is constructed using a recently proposed heuristic for the problem, [8]. The heuristic is based in maintaining the differences among accumulated weights within certain limits. The constructive heuristic is depicted in Algorithm 1, in which the function *OrderBins* orders the bins in a non-increasing order of accumulated weights, and the function *OrderQuestions* orders the questions in a non-decreasing order of the weights.

b/t	*1*	*2*	*3*	*4*	*5*
1	A (2)	B (8)	C (1)	C (4)	C (7)
2	B (4)	A (6)	A (7)	B (5)	B (6)
3	C (3)	C (4)	B (9)	A (8)	A (6)

b/t	*1*	*2*	*3*	*4*	*5*
1	C (2)	A (8)	A (1)	C (4)	B (7)
2	B (4)	B (6)	B (7)	A (5)	C (6)
3	A (3)	C (4)	C (9)	B (8)	A (6)

Fig. 1 Two examples of encoding. *Each letter* corresponds to a different questionnaire and the *number in parenthesis* represents the weight of the question

Algorithm 1: New constructive heuristic

```
for b := 1 → B do
    W_b = 0
end for
for t := 1 → T do
    OrderBins(W_b)
    OrderQuestions(η_rt)
    for r := 1 → B do
        Assign item r to bin r
        W_r := W_r + η_rt
    end for
end for
```

Note that the heuristic provides different solutions depending on the initial orderings of the sets. Accordingly, the initial population is created by using this constructive heuristic with different random orderings. A local search is also applied to each initial individual. The neighbourhood is defined by the removal from the solution of all of the items of a set, and their reintroduction using the assignment rule from Algorithm 1. This initialisation method generates near-optimal solutions, which the GA tries to combine into better solutions using an optimisation-based crossover operator.

2.3 Genetic Operators

Both the selection and replacement operators make use of standard tournament processes in which two individuals are randomly chosen and the best (worst) among them is used for crossover (replacement). The crossover operator is a combination of the classical n-point crossover method with an optimisation subproblem. The main idea is to aggregate the sets in accordance to the information present in the parents, and to solve (perhaps approximately) a reduced problem using the mixed zero-one integer linear formulation presented in Sect. 1. Similar approaches have been proposed by different authors for several problems, see [6].

The mechanism of this crossover operator is the following: first, the operator generates n different cutting points, $c_1, \ldots, c_n$, which are drawn from a uniform distribution $[1, T]$. Based on these cutting points, the chromosomes of the parents will be divided in n (if $c_n = T$) or $n+1$ segments. If the cutting points are ordered in increasing order $c_1 < c_2 < \cdots < c_n$, we can identify segment 1 with chromosomes $[1, c_1]$, segment 2 with chromosomes $[c_1 + 1, c_2]$, and so on until the last segment, which corresponds to $[c_{n-1} + 1, T]$ or $[c_n + 1, T]$.

These segments are then used to define the aggregated sets. An aggregated set is obtained by joining, for each questionnaire, all of the questions assigned to it (e.g. all questions assigned to questionnaire A will define question 1 in the aggregated

	$t' = 1$	$t' = 2$
$b' = 1$	15	11
$b' = 2$	21	15
$b' = 3$	8	10

	$t' = 1$	$t' = 2$
$b' = 1$	A'	A'
$b' = 2$	B'	C'
$b' = 3$	C'	B'

	$t = 1$	$t = 2$	$t = 3$	$t = 4$	$t = 5$
$b = 1$	A (2)	B (8)	C (1)	A (4)	C (7)
$b = 2$	B (4)	A (6)	A (7)	B (5)	A (6)
$b = 3$	C (3)	C (4)	B (9)	C (8)	B (6)

Fig. 2 Aggregated instance, solution for the aggregated problem and individual after disaggregation

set). In order to incorporate information from both parents, the chromosomes from parent 1 (parent 2) are used to aggregate odd-numbered (even-numbered) segments.

Figure 2 depicts an example using the individuals from Fig. 1 (parent 1 and parent 2, respectively). Suppose that a single cutting point, $c_1 = 3$, is used. The resulting aggregated questions for segment 1 are extracted from parent 1 as follows: question 1 corresponds to questionnaire A with a total weight of 15; question 2 corresponds to questionnaire B with weight 21 and question 3 to questionnaire C with weight 8. Similarly, the aggregated questions for segment 2 correspond to the questionnaires A, B and C from parent 2. The optimal solution to this problem is to assign together question 1 from the first aggregated set with question 1 of the second aggregated set, question 2 of the first set with question 3 of the second one, and question 3 of the first set with question 2 of the second one.

After disaggregation, the individual created by the operator is also depicted in Fig. 2. Its objective function value is 31, which improves the objective function of both parents.

Please note that we do not make use of a mutation operator, as this GA is designed as a method to intensify the search rather than as a diversification mechanism to explore large areas of the solution space.

3 Computational Experiments

To assess the quality of the proposed GA, the GA and the SA procedure proposed in [1] were programmed in C++. The IBM CPLEX solver was used to solve the formulation as well as the hybrid crossover subproblems. These procedures were tested on a randomly generated set of instances following the proposal from [1]. A total of 340 instances with a number of questions ranging from 300 to 6000, and a number of questionnaires ranging from 2 to 300 were constructed.

All of the algorithms were run for a maximum of 600 s. The default parameter configuration of CPLEX was used, with the exception of the relative and absolute gaps which were set to 0 in order to avoid the behaviour detected in previous experiments, see [1]. The parameters of the SA were identical to those proposed in [1]. Furthermore, if the SA terminates before the running time limit, the algorithm is restarted with a different random initial solution.

The parameters of the GA were chosen after tuning by hand. The population size was set to 50 individuals, one replacement is performed per iteration and a 3-point hybrid crossover is used. Additionally, in order to avoid CPLEX spending large amounts of time verifying optimality instead of improving the best known solution, a maximum running time for each execution of the hybrid crossover is imposed. This time limit is set to 3 s per crossover.

The results of the computational experiment show that CPLEX is to be the preferred method for instances with a small number of questions or questionnaires, as it is capable of finding the optimal solution. On the other hand, the GA outperforms the other methods for instances with many questions and questionnaires. Note that the SA outperforms the GA for small size instances, leading us to conjecture that some additional improvements, like the introduction of some form of mutation, may lead to an improvement in the proposed method.

References

1. Brusco, M.J., Köhn, H.F., Steinley, D.: Exact and approximate methods for a one-dimensional minimax bin-packing problem. Ann. Oper. Res. **206**, 611–626 (2013)
2. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, Reading (1989)
3. Holland, J.H.: Adaptation in Natural and Artificial Systems: an introductory analysis with applications to biology, control, and artificial intelligence. The University of Michigan Press, Oxford (1975)
4. Johnson, D.S., Garey, M.R., Graham, R.L., Demers, A., Ullman, D.: Worst-case performance bounds for simple one dimensional packing algorithms. SIAM. J. Comput. **3**, 299–325 (1974)
5. Martello, S., Toth, P.: Knapsack Problems: Algorithms and Computer Implementations. Wiley, Chichester (1990)
6. Talbi, E.G.: A taxonomy of hybrid metaheuristics. J. Heuristics **8**, 541–564 (2002)
7. Van der Linden, W.J.: Linear Models for Optimal Test Design. Springer, New York (2005)
8. Pereira, J., Vilà, M.: Variable neighborhood search heuristics for a test assembly design problem. Expert Syst. Appl. **42**(10), 4805–4817 (2015)

The Partial Choice Recoverable Knapsack Problem

Clément Lesaege and Michael Poss

1 Introduction

We study in this paper a variant of the knapsack problem. We are given a capacity c and a set $N = \{1, \ldots, n\}$ of available items. The weight and profit of item i are denoted by ω_i and p_i, respectively. Our objective is to select a subset of items $X \subseteq N$ of maximum profit knowing that k items will be removed from X (removed items count in the objective but not in the capacity limit). Namely, after X is fixed, an opponent chooses to remove from X a subset of Γ items, denoted by $\overline{X}$. Then, a subset of $l = k - \Gamma$ items of our choice, denoted by Z, is further removed from X. Hence, a solution is feasible if and only if for each set of items removed at the first stage, it is possible to remove l items at the second stage such as the knapsack capacity is not exceeded. This can be modeled as follows:

$$\begin{aligned} &\max_{X \subseteq N} \quad \sum_{j \in X} p_j \\ &\text{s.t.} \quad \max_{\substack{\overline{X} \subseteq X \\ |\overline{X}| \leq \Gamma}} \min_{\substack{Z \subseteq X \\ \overline{X} \subseteq Z \\ |Z| \leq k}} \sum_{j \in X \setminus Z} \omega_j \leq c. \end{aligned} \quad (1)$$

C. Lesaege
UMR CNRS 7253 Heudiasyc, Université de Technologie de Compiègne, Centre de Recherches de Royallieu, 60200 Compiègne, France
e-mail: clement.lesaege@etu.utc.fr

M. Poss (✉)
UMR CNRS 5506 LIRMM, Université de Montpellier, 161 rue Ada, 34392 Montpellier Cedex 5, France
e-mail: michael.poss@lirmm.fr

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_24

We call the above problem the Partial Choice Recoverable Knapsack Problem (PCKP), since a total of k items are removed including at most Γ items removed without possibility of choice. We show in the next section that the problem is a special case of the recoverable robust knapsack problem (RRKP) considered previously in the literature [1, 2]. While the complexity of the RRKP is still unknown, we show in Sect. 3 that PCKP is NP-hard in the weak sense and provide a dynamic programming algorithm for the problem, inspired by the dynamic programming algorithm for the robust knapsack problem [3]. We provide below a simple application example of the problem.

A shipping company has been requested to ship different kinds of cumbersome items that need whole dedicated vehicles to ship them. For each of these shipping requests, our company has an estimated delivery date. We define $N = \{1, \ldots, n\}$ as the set of these item requests. Each of these items requests has a weight ω_j determined by the quantity of labour needed to ship the item. The profit for accepting a request is p_j. The total quantity of labour available to ship the items before the estimated delivery date is c. We assume that up to Γ selected items will not be available for shipping on time (for example because the producers will be late in their production or companies will have mismanaged their stocks). Obviously, the items not available for shipping do not consume any labour available before the estimated delivery date. If our shipping company delivers too many items after the estimated delivery date, it will loose its reputation, but if only few of them are delivered late, the damage will be insignificant. Therefore, up to k items may be delivered after the estimated delivery date (whether the delay is due to the shipping company or the producer does not matter). Our goal is to determine which of the shipping requests should be accepted in order maximise the profit of our shipping company. The constraints impose that, no matter which items are not available for shipping, no more than k items will miss the delivery date.

2 Link with the Recoverable Robust Knapsack Problem

Büsing et al. [1] define the Recoverable Robust Knapsack Problem as follows.

Definition 1 Let N be a set of n items with profits p_i, nominal (or default) weight ω_i, and maximum deviation $\hat{\omega}_i$, $i \in N$. For a given $\Gamma \in \mathbb{N}$, the set $\mathscr{S}_\Gamma$ consists of all scenarios S which define a weight function $\omega^S : N \mapsto \mathbb{N}$ s.t. $\omega_i^S \in [\omega_i, \omega_i + \hat{\omega}_i]$ for all $i \in N$ and $|\{i \in N : \omega_i^S > \omega_i\}| \leq \Gamma$. For $k \in \mathbb{N}$ and a subset $X \subseteq N$ the recovery set $\mathscr{X}_X^k$ consists of all subsets of X with at least $|X| - k$ elements, i.e., $\mathscr{X}_X^k = \{X' \subseteq X : |X \backslash X'| \leq k\}$. Given a knapsack capacity $c \in \mathbb{N}$, the RRKP is to find a set $X \subseteq N$ with a maximum profit $p(X) := \sum_{j \in X} p_j$ s.t. for every scenario $S \in \mathscr{S}_\Gamma$ there exists a set $X' \in \mathscr{X}_X^k$ with $\omega^S(X') \leq c$.

We show next that the Partial Choice Recoverable Knapsack Problem is a particular case of the Robust Recoverable Knapsack Problem.

Theorem 1 *Let* $(c, p, \omega, \hat{\omega}, \Gamma, k)$ *describe an instance of the RRKP and let* $(c, p, \omega, \Gamma, k)$ *be an associated instance for the PCKP. If the instance satisfies conditions (i)* $\Gamma \leq k$*, and (ii)* $\forall i, j \in N : \omega_i \leq \omega_j + \hat{\omega}_j$*, then both problems have the same optimal solution* $X \subseteq N$*.*

Proof Let $X \subseteq N$ be a potential solution for the RRKP and let us denote by Y and $\overline{X}$ the sets of items that are removed from X and deviate, respectively. We see that X is feasible for the RRKP if it satisfies the following constraint:

$$\sum_{i \in X} \omega_i + \max_{\substack{\overline{X} \subseteq X \\ |\overline{X}| \leq \Gamma}} \left(\sum_{i \in \overline{X}} \hat{\omega}_i - \max_{\substack{Y \subseteq X \\ |Y| \leq k}} \left(\sum_{i \in Y} \omega_i + \sum_{i \in \overline{X} \cap Y} \hat{\omega}_i \right) \right) \leq c. \quad (2)$$

Because of assumption (ii), any solution of the inner maximization in the left-hand side of (2) satisfies $Y \subseteq \overline{X}$. Said differently, we must always remove the items that deviate. Denoting $Y^* = Y \cap (X \setminus \overline{X})$ we must withdraw from the left-hand side of (2) the Γ items that deviate as well as the other items removed, denoted by Y^*. Replacing the inner maximization with the minimization of its opposite, constraint (2) becomes:

$$\max_{\substack{\overline{X} \subseteq X \\ |\overline{X}| \leq \Gamma}} \min_{\substack{Y^* \subseteq X \setminus \overline{X} \\ |Y^*| \leq k - \Gamma}} \sum_{i \in X \setminus (\overline{X} \cup Y^*)} \omega_i \leq c. \quad (3)$$

By setting $Z = \overline{X} \cup Y^*$, we obtain to the formulation of the Partial Choice Recoverable Knapsack Problem (1).

This equivalences implies that we can solve the PCKP as a RRKP by setting appropriate values for $\hat{\omega}_i$ and using the integer linear programming formulation from [1]. Nevertheless, we show in this paper how to develop a more efficient way to solve the problem.

3 Dynamic Programming Algorithm

When $k = \Gamma$, the next result shows that the PCKP yields a robust knapsack problem. The proof is omitted due to the length limitation.

Theorem 2 *When* $k = \Gamma$*, the PCKP can be solved by using the dynamic programming algorithm of the Robust Knapsack Problem by setting the deviations to* $\hat{\omega}_i = -\omega_i$*.*

We show in the rest of the section how to extend the dynamic programming algorithm from [3] to the general case of PCKP. In view of Theorem 2, we can assume that $k > \Gamma$. This means we can plan to remove $l = k - \Gamma$ items from the knapsack without any particular condition and Γ items will be removed without possibility of choice. A new way to write constraint is:

$$\max_{\substack{\overline{X}\subseteq X\\ |\overline{X}|\leq \Gamma}} \min_{\substack{Y\subseteq X\\ |Y|\leq l}} \sum_{j\in X\setminus(Y\cup\overline{X})} \omega_j \leq c.$$

We use an equivalent formulation of the constraint:

$$\forall \overline{X} \subseteq X, |\overline{X}| \leq \Gamma : \exists Y \subseteq X, |Y| \leq l \text{ such as } \sum_{j\in X\setminus(Y\cup\overline{X})} \omega_j \leq c. \quad (4)$$

We order items by decreasing weight ($i > j \Rightarrow \omega_i \leq \omega_j$). From now, we will consider that items are sorted this way. In the worst case scenario, $\overline{X}$ contains the items with the lowest weight. This is formalized in the next lemma.

Lemma 1 *Let* $\overline{X} \subseteq X$ *and* $Y \subseteq X$ *such that* $\sum_{j\in X\setminus(Y\cup\overline{X})} \omega_j \leq c$. *For any pair of indices such that* $i > j$ *and* $j \in \overline{X}$, *let us define* $\overline{X'} = \overline{X} + \{i\} - \{j\}$. *We can construct a set* $Y' \subseteq X$ *such as* $\sum_{j\in X\setminus(Y'\cup\overline{X'})} \omega_j \leq c$.

Proof If $i \notin Y$, we set $Y' = Y$. Since $i > j$ implies that $\omega_i \leq \omega_j$, $\sum_{j\in X\setminus(Y'\cup\overline{X'})} \omega_j \leq \sum_{j\in X\setminus(Y\cup\overline{X})} \omega_j \leq c$ and the constraint is still satisfied. If $i \in Y$, we set $Y' = Y - \{i\} + \{j\}$, so $\left(Y' \cup \overline{X'}\right) = \left(Y \cup \overline{X}\right)$ and the constraint is still satisfied.

Theorem 3 *Partial Choice Recoverable Knapsack Problem is NP-hard in the weak sense and can be solved in* $\mathcal{O}\left(n\left(\Gamma c + (k-\Gamma) + \ln(n)\right)\right)$.

Proof From Lemma 1, we see that Y can be assumed to contain the items with the highest weights; if some of these were removed by the opponent (put in $\overline{X}$), we could remove any other items instead the constraint would still be satisfied. Thus, we can reformulate (4) as follows:

$$\begin{gathered} Y = \{l \text{ first items of } X\} \\ \forall \overline{X} \subseteq X\setminus Y, |\overline{X}| \leq \Gamma : \sum_{j\in X\setminus(Y\cup\overline{X})} \omega_j \leq c. \end{gathered} \quad (5)$$

We will build our algorithm in two phases. The first one will be to determine the profit we can get from items in Y (we will call it first stage profit) and a second one to determine the profit we can get from items not in Y (we will call it second stage profit). The two phases are linked: the items which may appear in $\overline{X}$ depend on items in Y. Since $Y = \{l \text{ first items of } X\}$ we can partition N into two subsets $\{0, \ldots, t\}$ and $\{t+1, \ldots n\}$ where the first one is the set of items that can be put in Y and the second one is the set of items that cannot. To explore all relevant possibilities, we explore all possible partitions of N into two sets of consecutive integers. There are $n+1$ possible partitions.

For the first stage profit, we assume that the set of items in Y is restricted to $\{0, \ldots, t\}$. Let $y(t, s)$ denote the maximum first stage profit considering t items and $l = s$. We compute $y(t, s)$ as in a regular knapsack problem considering that the capacity is $l = k - \Gamma$ and the weights of each items is 1. We compute the second stage profit $R(t)$ by applying the Robust Knapsack dynamic programming algorithm on the subset $\{t + 1, \ldots, n\}$ since $k = \Gamma$ for that subset. The maximum profit is $\max_{t \in N}(y(t, l) + R(t))$. Putting less than l items in the knapsack at the first stage cannot increase the profit, therefore, we have explored all possibilities.

We sort items by decreasing weights ($i > j \Rightarrow \omega_i \leq \omega_j$) and we compute the first stage profits by the following dynamic programming recursions:

$$y(j, s) = \max(y(j-1, s-1) + p_j, y(j-1, s)) \tag{6}$$

The initialization values are $y(j, 0) = 0$ for $j = 0, \ldots, t$ and $y(0, s) = 0$ for $s = 0, \ldots, l = k - \Gamma$.

To compute the second stage profit, the naive solution would be to simply apply the Robust Knapsack dynamic programming algorithm to the subset $\{i + 1, \ldots, n\}$ for each value of i. However, considering that we use a dynamic programming algorithm there is a more efficient solution. We create a Robust Knapsack problem by the setting the deviation to $\hat{\omega}_i = -\omega_i$. We need to sort the items by increasing deviations (remember that we are working with negative deviations). To do so, we just have to reverse the indexes ($j' = n - j + 1$). We compute all second stage profit by the following dynamic programming recursion given by Monaci et al. [3]:

$$\begin{aligned} \bar{z}\left(d, s, j'\right) &= \max(\bar{z}\left(d, s, j'-1\right) + p_{j'}, \bar{z}\left(d - \left(\omega_{j'} + \hat{\omega}_{j'}\right), s-1, j'-1\right)) \\ &\qquad \text{for } d = 0, \ldots, c, s = 1, \ldots, \Gamma, j' = 1, \ldots, n, \\ z\left(d, j'\right) &= \max(z\left(d, j'-1\right), z\left(d - \omega_{j'}, j'-1\right) + p_{j'}) \\ &\qquad \text{for } d = 0, \ldots, c, j' = \Gamma + 1, \ldots, n, \end{aligned} \tag{7}$$

We initialise $\bar{z}(d, s, 0) = -\infty$ for $d = 0, \ldots, c, s = 1, \ldots, \Gamma$. We set $z(0, 0, 0) = 0$ and we link the two arrays by $z(d, \Gamma) = \bar{z}(d, \Gamma, \Gamma)$ for $d = 0, \ldots, c$.

We notice that $\omega_{j'} + \hat{\omega}_{j'} = 0$ and we remove this term from the recursion of $\bar{z}$. The optimal value of the second stage profit considering only the first q items is:

$$z^*(q) = \max \begin{cases} \max_{d=1,\ldots,c} z(d, q) \\ \max_{\substack{d=1,\ldots,c, \\ s=1,\ldots,\Gamma-1}} \bar{z}(d, s, q) \end{cases}$$

Since we have reversed the indexes for the second stage, $z^*(n - t + 1)$ is the maximum profit for the Robust Knapsack problem of subset $\{t + 1, \ldots n\}$. It results that $R(t) = z^*(n - t + 1)$. We have computed the maximum profit of all second stage subsets using only one call of the Robust Knapsack dynamic programming algorithm.

The first sorting algorithm is in $\mathscr{O}(n\ln(n))$, reversing indexes is in $\mathscr{O}(n)$. The computation of the first stage solution is in $\mathscr{O}(l\,n) = \mathscr{O}((k-\Gamma)\,n)$. The use of the Robust Knapsack dynamic programming algorithm on already sorted data is in $\mathscr{O}(\Gamma\,c\,n)$. The resulting complexity is $\mathscr{O}(n(\Gamma c + (k-\Gamma) + \ln(n)))$. We have found a pseudo-polynomial algorithm, so the problem is at worst weakly NP-hard. By setting $k = \Gamma = 0$, we can solve a regular knapsack problem which is a weakly NP-hard problem, so our problem is at least weakly NP-hard. In conclusion the Partial Choice Recoverable Knapsack Problem is NP-hard in the weak sense.

4 Numerical Experiments

Table 1 Highest value of n such as the problem is solved in less than 30 s with $k = 2\Gamma$ and $c = 200$

Γ in % of n	Dynamic programming	ILP
1	12,400	235
2	9200	233
5	5800	77
10	4200	57

To test the practical efficiency of our algorithm, we generate random instances where ω_j is randomly chosen in $[0, c]$ and p_j in $[1, 100]$. The algorithm is coded in C++ and run on a PC Intel CPU@2.53 GHz with 2 GB of available RAM. We compare our dynamic programming algorithm with the MILP provided by Büsing et al. in [1] for the RRKP. In view of Theorem 1, one can also use their formulation to solve the PCKP. The MILP is solved by Xpress Optimization Suite. We determine the largest instance that can be solved in less than 30 s. The results from Table 1 show that the dynamic programming is more efficient by order of magnitude. Larger instances could not be solved due to our space limitations.

References

1. Büsing, C., Koster, A.M., Kutschka, M.: Recoverable robust knapsacks: *gamma*-scenarios. In: Proceedings of INOC 2011, International Network Optimization Conference. Lecture Notes on Computer Science, vol. 6701, pp. 583–588. Springer, Heidelberg (2011). http://dx.doi.org/10.1007/978-3-642-21527-8_65
2. Büsing, C., Koster, A.M.C.A., Kutschka, M.: Recoverable robust knapsacks: the discrete scenario case. Optim. Lett. **5**(3), 379–392 (2011)
3. Monaci, M., Pferschy, U., Serafini, P.: Exact solution of the robust knapsack problem. Comput. Oper. Res. **40**(11), 2625–2631 (2013)

Stopping Times for Fractional Brownian Motion

Alexander V. Kulikov and Pavel P. Gusyatnikov

1 Introduction

Modern financial mathematics is based on the theory of semimartingales and Markov processes. Nevertheless there is a process of fractional Brownian motion introduced by Mandelbrot and van Ness in [4] which is often used in practice. There is no equivalent martingale measure for models which use fractional Brownian motion (see [1]). Cheridito in [1] showed that even if fractional Brownian motion market assumes arbitrage strategies these strategies cannot be realized in practice since there is always a time delay between transactions in practice. Guasoni in [3] considered that if transaction costs exist then an opportunity of arbitrage vanishes as well.

A problem of optimal stopping time is of our interest. In financial sense this problem can be interpreted as follows: if an investor has a set of assets he has to make a decision: at which time he should sell these assets. In paper [5] a problem of optimal stopping time has been considered as a problem of maximization expectation value of a utility function. A process considered was a process of Brownian motion with a drift. The solution of a problem derived in that paper can be represented as "Buy and Hold" rule.

In this paper we will show that for fractional Brownian motion "Buy and Hold" rule cannot be applied. We will discuss a class of stopping times which can be claimed to be optimal and easy to model.

A.V. Kulikov PhD
Moscow Institute of Physics and Technology, Moscow, Russia
e-mail: avkulikov15@gmail.com

P.P. Gusyatnikov (✉)
Moscow Institute of Physics and Technology, Moscow, Russia
e-mail: gusyatnikov.pavel@gmail.com

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_25

The paper is organized as follows: in Sect. 2 we will discuss properties of fractional Brownian motion, a problem of optimal stopping time will be introduced and some stopping time classes will be shown. In Sect. 3 we will show an example of non optimality of these stopping times and results of numerical modeling showing non triviality of these stopping times.

2 Optimal Stopping Problem

First of all we should introduce an optimal stopping problem. Consider an asset which price changes corresponding to a stochastic process X. An owner of this asset should sell this asset till time T in the best way. This problem is called optimal stopping time problem. Mathematically this problem is formulated as follows: we should find a time τ^*:

$$\tau^* = argmax(\mathsf{E}U(X_\tau)), 0 \leq \tau \leq T$$

where U(x) is an utility function, τ is a stopping time.
In [5] authors have shown that for classical Brownian motion a solution of this problem can be represented as "Buy and hold" rule. An owner of an asset should sell it either at time t = 0 or at time t = T. In this paper we will show that for fractional Brownian motion a solution of this problem is more complex.

Definition 1 (see [4]) *Fractional Brownian motion* is a gaussian stochastic process $B_H(t)$ with the following properties:

- $B_H(0) = 0$ and $\mathsf{E}[B_H(t)] = 0$;
- $cov(B_H(t), B_H(s)) = \frac{1}{2}(t^{2H} + s^{2H} - |t-s|^{2H})$.

Parameter H is called *Hurst parameter*.

Remark 1 If $H = \frac{1}{2}$ then this process is a classical Brownian motion.

Remark 2 Having $H < \frac{1}{2}$ this process has negative autocorrelation. Having $H > \frac{1}{2}$ this process has a positive autocorrelation.

Remark 3 Fractional Brownian motion has a self-similarity property with Hurst parameter equal to H. This means that if a process $B_H(t)$ is a process of fractional Brownian motion, then the following processes have the same distributions

$$\{B_H(at) : 0 \leq t < \infty\} \stackrel{d}{=} \left\{a^H B_H(t) : 0 \leq t < \infty\right\}.$$

Using Remark 3 and a property of continuity of fractional Brownian motion we can study properties of this process for discrete times and then rescale it for any time interval.

There are several different ways of fractional Brownian motion modeling. In this paper we will use a model proposed by Dieker in [2]. Fractional Brownian motion has the following covariation function:

$$\gamma(s) = cov(X_H(t), X_H(t+s)) = \frac{1}{2}(|s-1|^{2H} + |s+1|^{2H} - 2|s|^{2H}).$$

Let $\Gamma(n) = \{\gamma(i-j)_{(i,j=1,n)}\}$ be a covariation matrix, and $c(n)$ is a vector of size $(n+1)$, where $c(n)_k = \gamma(k+1)$. Then $X_{n+1} = B_h(n+1) - B_h(n)$ is a random value with normal distribution with expectation μ_n and variance σ_n^2, where $\mu_n = d(n)^T \begin{pmatrix} X_n \\ X_{n-1} \\ \vdots \\ X_0 \end{pmatrix}$, $\sigma_n^2 = 1 - c(n)^T \Gamma(n)^{-1} c(n)$. Dieker shows that there is an iterative algorithm for modeling μ_n and σ_n^2 with no need of matrix inversion as well.

An optimal stopping time problem is a following problem:

$$f(h) = \sup_{\tau \le 1} \mathsf{E} B_H(\tau), \tau(h) = \arg\max_{\tau \le 1} \mathsf{E} B_H(\tau).$$

Using self similar property this problem is equivalent to the following problem:

$$f_N(h) = \frac{1}{N^h} \sup_{\tau \le N} \mathsf{E} \sum_{i=0}^{\tau} X_H(i), \tau_N(h) = \frac{1}{N^h} \arg\max_{\tau \le N} \mathsf{E} \sum_{i=1}^{N} X_H(i).$$

In this problem a following class of stopping times τ is considered: $\tau_c = \frac{1}{N^h} min(t : \mu_t < c(N-t) * t^H)$, where $\mu_t = \mathsf{E}(X_h(t+1)|X_h(1), \ldots, X_h(t))$.

Example 1 Consider the case of $c = 0$. From financial point of view this means that we don't sell assets till the moment when the next increment has a negative average.

Here we have following limit cases:

- $H = 0.5$. In this case $\mu_t = 0 \ \forall t$, i.e. $\mathsf{E} B_\tau = 0$.
- $H = 1$ In this case $X(t) = X(1) \ \forall t$. This means

$$\sup_{\tau \in [0,1]} \mathsf{E} B_\tau = \int_0^\infty x \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} dx = \frac{1}{\sqrt{2\pi}},$$

but the optimal moment cannot be reached.

It can be shown that

$$\mathsf{E}(\mu_{n+1}|F_n) = \mathsf{E}(d_0^{n+1}X_{n+1}|F_n) + \sum_{i=1}^{n+1} d_{n-i+1}^{n+1}X_i = d_0^{n+1}\mu_n + \sum_{i=1}^{n+1} d_{n-i+1}^{n+1}X_i,$$

$$\mathsf{E}(\mu_{n+2}|F_n) = \mathsf{E}(d_0^{n+2}X_{n+2}|F_n) + \mathsf{E}(d_1^{n+2}X_{n+1}|F_n) + \sum_{i=2}^{n+2} d_i^{n+2}X_{n-i+2} =$$

$$= d_0^{n+2}\mathsf{E}(\mu_{n+1}|F_n) + d_1^{n+2}\mu_n + \sum_{i=2}^{n+2} d_i^{n+2}X_{n-i+2}.$$

And for any k we have $\mathsf{E}(\mu_{n+k}|F_n) = \sum_{j=0}^{k-2}\mathsf{E}(d_j^{n+k}X_{n+k-j}|F_n) + d_{k-1}^{n+k}\mu_n + \sum_{i=k}^{n+k} d_i^{n+k}X_{n-i+k}$, i.e. an average for any future values of fractional Brownian motion according to the information available for current moment is linear combination of values of discretization of fractional Brownian motion.

Example 2 We consider the following stopping times class as τ:

$$\tau_k = \frac{1}{N^h}min(n : \mu_n < 0, \sum_{i=0}^{j} E(\mu_{n+i}|F_n) < 0, \forall j \leq k).$$

From financial point of view this means that we do not sell assets if there is a tendency to growth for at least any interval from 0 to k.

3 Modeling Results

In this section we will consider modeling results for different stopping times.

On (Fig. 1) modelling results of τ_c for different c are presented. As we see, for quite small negative c we have quite good results for $H > 0.5$. As we see when H tends to 1 we have a result that was shown before: $\frac{1}{\sqrt{2\pi}}$. For $H < 0.5$ we do not consider small H because we have problems with continuity of a process. But we see that for $H < 0.5$ we have non trivial results as well. As we see when $H > 0.5$ stopping time corresponding to $c = 0$ is not optimal.

Example 3 It can be shown that $E(\mu_{n+k}|F_n)$ can be positive even if $E(\mu_n|F_n) < 0$, where F_n is a filtration of $X_0, X_1, \ldots, X_n$. We shall assume $H = 0.7$,

$$X_0 = 1, X_1 = 1, X_2 = 2, X_3 = -1$$

$$d^4 = (0.278596, 0.0728796, 0.0540914, 0.0507365)$$

$$d^5 = (0.276542, 0.07069, 0.0511412, 0.0394588, 0.0404806)$$

$$\mu_3 = -0,0280089 < 0$$

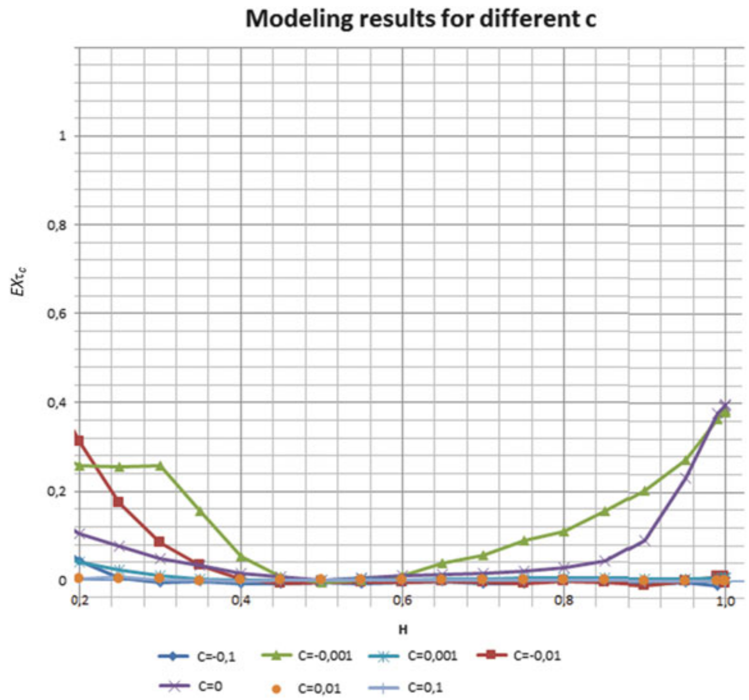


Fig. 1 Modeling results for τ_c for different c. T=1000. On X-Axis—H, on Y-axis—corresponding values of EX_{τ_c} for different c. Since it's difficult to model fractional Brownian motion when H is small we discuss only $H > 0.2$

But in this case,

$$E(\mu_4|X_0, X_1, X_2, X_3) = 0,0526450 > 0$$

Moreover,

$$E(\mu_4|X_0, X_1, X_2, X_3) > -\mu_3$$

This means that there exists stopping time which is even better than $\tau = \frac{1}{N^h} min(t : \mu_t < 0)$.

We see that proposed stopping times give us non trivial results as well. The results of modeling of τ_k can be shown on the previous figure (Fig. 2). As we see on it for $H > 0.5$ we have bigger values of EB_{τ_k} for bigger k. When H tends to 1 we have the same value that we have got for τ_c. We see that for $H > 0.5$ τ_k gives better results than τ_c and it can claim to be optimal.

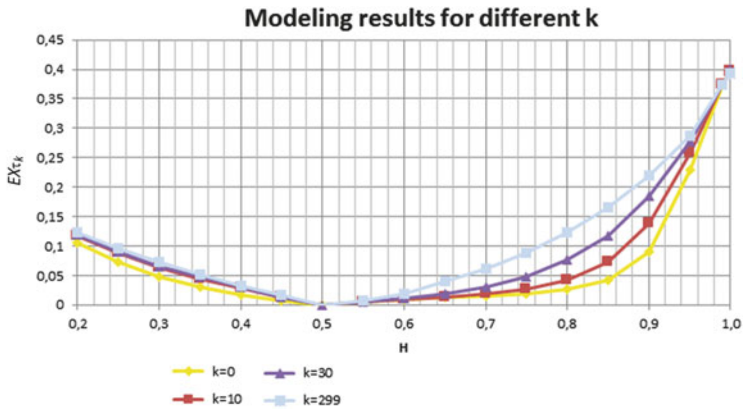


Fig. 2 Modeling results for τ_k for different k ($N = 1000$). On X-Axis—H, on Y-axis—corresponding values of EX_{τ_k} for different k. Since it's difficult to model fractional Brownian motion when H is small we discuss only $H > 0.2$

References

1. Cheridito, P.: Arbitrage in fractional brownian motion models. Finance Stochast. **7**(4), 533–553 (2003)
2. Dieker, T.: Simulation of fractional brownian motion. Master's thesis, Department of Mathematical Sciences, University of Twente, Enschede (2004)
3. Guasoni, P.: No arbitrage under transaction costs, with fractional brownian motion and beyond. Math. Financ. **16**(3), 569–582 (2006)
4. Mandelbrot, B., van Ness, J.W.: Fractional brownian motions, fractional noises and applications. SIAM Rev. **10**(4), 422–437 (1968)
5. Shiryaev, A., Xu, Z., Zhou, X.Y.: Thou shalt buy and hold. Quant. Finance **8**(8), 765–776 (2008)

An Empirical Design of a Column Generation Algorithm Applied to a Management Zone Delineation Problem

Víctor M. Albornoz and Linco J. Ñanco

1 Introduction

The problem of delineating site-specific management zones in agricultural fields arises in the context of precision agriculture, where the control of variability in soil properties is essential to increase productivity and crop quality. Dividing a field into high internal homogeneity zones with respect to any soil property (pH, organic material, phosphorus, etc.) allows the farmer to face this variability. Furthermore, establishing a rectangular management zone partition provides practical advantages with respect to the use of agricultural machinery.

This paper presents different strategies based on the column generation technique to solve the problem proposed in [2], focused on the delineation of rectangular management zones, considering a fixed internal homogeneity level to solve the problem, measured by the relative variance of partition criterion used to assess the efficiency of the field division (see [6]).

Hereinafter, the problem discussed is considered as: given a set of sample points in a field, $S = \{1, \ldots, N\}$, and a set of potential quarters $Z = \{1, \ldots, K\}$, where each potential quarter covers a subset of these sample points, find the subset of Z with the minimum number of elements, which is a partition of field, with a given maximum relative variance. The following parameters are defined to introduce the solved model: parameter c_{zs}, $s \epsilon S$, $z \epsilon Z$ indicates whether quarter z covers sample point s or not; n_z, $z \epsilon Z$ indicates the number of sample points considered in quarter z; σ_z^2, $z \epsilon Z$ is the variance of quarter z with respect to the soil property analyzed; σ_T^2 is the total variance of field with respect to soil property analyzed; and LS is the

V.M. Albornoz • L.J. Ñanco (✉)
Departamento de Industrias, Universidad Técnica Federico Santa María, Av. Santa María 6400, Santiago, Chile
e-mail: victor.albornoz@usm.cl; linco.nanco@alumnos.usm.cl

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_26

maximum number of quarters to set in a partition. While, the decision variable of the problem is q_z, a binary variable indicating whether potential quarter $z \epsilon Z$ is part of the partition or not. Accordingly, the model will be:

$$\min \sum_{z \epsilon Z} q_z \tag{1}$$

$s.t.$

$$\sum_{z \epsilon Z} c_{zs} q_z = 1 \ , \ \forall s \epsilon S \tag{2}$$

$$\sum_{z \epsilon Z} q_z \leq LS \tag{3}$$

$$\sum_{z \epsilon Z} \left[(n_z - 1)\, \sigma_z^2 + (1 - \alpha)\, \sigma_T^2 \right] q_z \leq \sigma_T^2 N\, (1 - \alpha) \tag{4}$$

$$q_z \epsilon \{0, 1\} \ , \ \forall z \epsilon Z \tag{5}$$

The objective function (1) minimizes the amount of potential quarters which are part of a partition. Constraint (2) is typical in partition models and ensures each soil sampling point to be covered by one single potential quarter. Constraint (3) establishes an upper bound to the number of quarters or zones where the field is divided, which is also needed to compute a lower bound on the optimal solution of the linear relaxation of the problem, as will be explained in Sect. 2. Constraint (4) is a linear equivalent rearrangement of a nonlinear constraint that establishes an upper bound $(1 - \alpha)$ on the relative variance of field delineation, with a given value of $\alpha \epsilon\ [0, 1]$ (see [6]). Finally, constraint (5) states that each decision variable q_z is binary. Herein this paper is organized as follows: Sect. 2 presents the algorithmic strategies developed, Sect. 3 presents the results obtained from several computational experiences, while Sect. 4 presents the conclusions of this research.

2 Algorithmic Strategies

The column generation method is a technique used in linear programming for solving problems with a large number of variables and a relatively small number of restrictions, which also have been successfully used in many integer programming problem solving. Several uses of this technique are referred to by Desaulniers et al. [3]. For more details about this method refer e.g. to [5]. Regarding the extension of this technique to integer programming problems, see [1] and [8], and particularly, for in set partitioning problems, see [7]. To the best of our knowledge, this is the first time that a column generation algorithm has been used to solve a management zone delineation problem.

Model (1)–(5) has set a partitioning problem structure with a formulation making it suitable to the application of the column generation method. So, linear relaxation of this model is considered as the Master Problem (MP) in the column generation scheme. Notice that each of the N sample points could be represented by the ordered pair (i,j) where $i = 1,..,n$ and $j = 1,..,m$. Therefore, the MP has a total of $n(n+1)m(m+1)/4$ variables and $nm+2$ constraints, for a rectangular field represented by $N = nm$ sample points, so the number of potential quarters grows polynomially with the number of sample points. Therefore, the complete enumeration of all the potential quarters is feasible but computationally expensive and it unnecessarily increases the problem size. It is interesting, then, to explore a more efficient strategy for solving the Master Problem (1)–(5).

2.1 The Pricing Problem

As usual in column generation, the pricing problem aims to verify the optimality of the Reduced Master Problem (RMP) by solving the minimization of the reduced cost function represented in terms of the set of potential quarter decision variables, whose optimal solution will be added to the RMP in case of a negative optimal value function. The potential quarter obtained should be rectangular shaped and must meet a set of sample points adjacent to each other. With this in mind, the pricing problem (SP) variable is x_s, a binary variable whose value is 1 if sample point s is covered by the new potential quarter, and 0 if not. The problem parameters correspond only to dual variable values p_s, ω and π of constraints (2)–(4) from RMP, respectively. SP can be represented as follows:

$$\min 1 - \sum_{s\epsilon S} p_s x_s - \omega - \pi \left[(n_z - 1)\, \sigma_z^2(x) + (1-\alpha)\, \sigma_T^2 \right] \tag{6}$$

s.t.

$$x_s \epsilon X \tag{7}$$

$$x_s \epsilon \{0, 1\} \ , \ \forall s \epsilon S \tag{8}$$

In the formulation above, target function (6) also includes function $\sigma_z^2(x)$, corresponding to the variance calculation of selected sample points, where the value of each sample point is d_s:

$$\sigma_z^2(x) = \frac{\sum_{s\epsilon S} x_s d_s^2}{\sum_{s\epsilon S} x_s} - \left(\frac{\sum_{s\epsilon S} x_s d_s}{\sum_{s\epsilon S} x_s} \right)^2 \tag{9}$$

2.2 Column Generation Strategies

Starting a column generation algorithm requires an initial RMP, that is, an initial set of columns and the corresponding variables. In this case, an initial set of potential quarters (q_z) each covering a single sample point, which means $\sigma_z^2 = 0$, provides an initial feasible solution. We also set $LS = N$, which is its trivial value.

Considering this initial RMP, for each iteration of the column generation algorithm, four different strategies to explore the pricing problem feasible solutions and identify new columns to be added to the RMP are proposed. Strategies I and II avoid solving the pricing problem (6)–(8) to optimality by iteratively fixing a size for the potential quarters to be considered (determined by an adjacent number of rows and columns), and then enumerating them and evaluating their reduced costs according to the objective value function (6). If a potential quarter with negative reduced cost is identified, its constraint coefficients are saved in order to be included into RMP up to a predefined number of negative potential quarters per iteration. The difference between these two strategies is that in Strategy I potential quarters are generated in ascending size order, while in Strategy II they are generated in descending size order. The algorithm stops after covering all possible quarter sizes and then solving the RMP as an integer program, where parameter LS is set with a slightly higher value than the best objective value reached by the application of the column generation algorithm.

On the other hand, Strategies III and IV are designed to improve the solution obtained by applying first Strategy I and II, respectively, and then by solving to optimality the RMP in the next iterations according to the column generation method. The difference between Strategies III and IV is that in the former only the column with the lowest reduced cost is added to the RMP at each iteration, while in Strategy IV a set of columns (including the optimal one) is added to the RMP, controlling the number of columns added at each iteration. Strategies III and IV use an optimality gap as a termination criteria to avoid the well known tailing-off effect, which appears as a slow convergence towards the MP optimal solution almost reached. Therefore, the duality based lower bound presented in [4] is used, specially designed for Set Partitioning formulations. To derive this duality based lower bound we need to add a redundant constraint that sets an upper bound on the sum of the variables (q_z), that justifies the use of constraint (3) in model (1)–(5). If we consider a parameter $\delta \leq SP^*$, where SP^* is the optimal value of SP, the lower bound can be calculated according to the following expression:

$$LB = \sum_{s \epsilon S} p_s + \sigma_T^2 N (1 - \alpha) \pi + LS\delta \tag{10}$$

During the algorithm execution, if the gap between the best upper and lower bounds is lower than a preset limit, the algorithm ends and an integrality condition is imposed on variables of the resulting RMP, which is solved to optimality by using solver CPLEX 12.4. This procedure does not guarantee the optimal solution for the MP. However, its quality level can be known by rounding up the lower bounds obtained from this algorithm. Section 3 presents results using the different strategies.

3 Computational Results

To compare the algorithmic strategies described in Sect. 2, computational experiments were carried out with randomly generated data sets, which represented 10 different instances with a minimum of 42 and a maximum of 900 sample points. Algorithms were coded in AMPL, both linear relaxations and integer linear problems were solved by using solver CPLEX 12.4 and were carried out on Intel Core i5-2450M 2.50 GHz with 8 GB of RAM.

Table 1 shows the set of different instances that we considered when studying the performance of the different strategies. It also shows the optimal value of model (1)–(5) for the first six instances of the problem. For the larger instances it was not possible to get an optimal solution after 50,000 s of running, considering that this time also included the complete generation of all possible potential quarters for solving (1)–(5) to optimality.

Tables 2 and 3 show the results for Strategies I and II. The UB column presents the best solution reached for the linear relaxation of the problem, while IS shows the objective value obtained by using CPLEX 12.4 after imposing the integrality condition. The #col column indicates the number of variables generated at the end of the algorithm. The gap column presents the difference between the upper bound reached by the application of Strategies I or II and the best known lower bound for each instance of the problem. Regarding the time, the results favor Strategy I where the potential quarters are considered in descending size order, however, looking at the upper bound that each one obtained, it is consistently worse than the one obtained in ascending size order. It means a significant negative impact in computational times (not shown in these results) when we implement Strategies III

Table 1 Instances and optimal value of problem (1)–(5)

Instance	N	K	Time (s)	Optimal value
1	42	588	0.48	10
2	100	3025	11.52	22
3	150	6600	106.14	24
4	225	14,400	1468.62	34
5	300	25,200	8761.71	47
6	400	44,100	47,654.8	58
7	500	68,250	–	–
8	600	97,650	–	–
9	750	151,125	–	–
10	900	216,225	–	–

Table 2 Results obtained for Strategy I

Instance	#col	UB	Gap	IS	Time (s)
1	148	9.07	0.02	10	0.64
2	401	23.59	0.13	24	2.12
3	609	34.61	0.46	35	5.41
4	1109	36.88	0.11	37	25.63
5	1470	56.3	0.23	57	58.67
6	2101	68.48	0.20	69	152.44
7	2443	98.37	0.29	99	248.14
8	3048	109.93	0.30	110	463.67
9	3829	138.2	0.31	139	891.75
10	4651	144.99	0.17	146	1540.15

Table 3 Results obtained for Strategy II

Instance	#col	UB	Gap	IS	Time (s)
1	161	8.93	0.00	10	0.89
2	433	22.38	0.07	23	2.96
3	726	25.12	0.06	26	10
4	1178	34.99	0.05	36	33
5	1730	47.09	0.03	48	99.96
6	2330	59.19	0.03	60	228.17
7	3115	80.75	0.06	82	482.42
8	3738	89.94	0.06	91	720.72
9	4916	112	0.06	113	1681.48
10	5198	134.3	0.08	135	2296.13

and IV using Strategy I, resulting in a higher algorithm end time. Therefore, only Strategy II was used with Strategies III and IV.

Table 4 shows the results obtained by using Strategy III, while Table 5 corresponds to Strategy IV. In these tables, the columns from left to right represent: the solved instance, #col is the number of total variables generated at the end of the algorithm, UB and LB are upper and lower bounds on the optimal value of the linear relaxation of the problem; gap % is the percentage gap between UB and LB , with tolerance at 2 % to complete algorithms, IS is the objective function value getting the last iteration of the algorithm when we solved the Reduced Master Problem as an integer program; ISLB is the lower bound for the Master Problem obtained from LB rounded upwards, so when ISLB is equal to IS, the optimal solution is observed and finally, Time is the completion time of the algorithm in seconds.

Results indicate that Strategy IV had better performance in time and determined the best upper and lower bounds on the optimal value of the linear relaxation of model (1)–(5). Moreover, we can observe that in nine of the ten instances Strategy IV reached the optimal solution of the problem.

Table 4 Results obtained for Strategy III

Instance	#col	UB	LB	% gap	IS	ISLB	Time (s)
1	172	8.89	8.89	4.50E−14	10	9	1.21
2	479	21.08	20.68	1.9	22	21	8.28
3	810	23.77	23.32	1.9	24	24	33.79
4	1317	33.24	32.62	1.84	34	33	130.95
5	1838	46.62	45.82	1.73	47	46	255.14
6	2579	57.52	56.41	1.94	58	57	934.17
7	3277	76.51	75.2	1.72	77	76	1295.44
8	4172	84.56	82.97	1.87	85	83	4254.71
9	5550	106.59	104.58	1.89	107	105	10,766.00
10	6095	124.33	122.04	1.84	125	123	19,268.80

Table 5 Results obtained for Strategy IV

Instance	#col	UB	LB	% gap	IS	ISLB	Time (s)
1	188	8.89	8.89	4.50E−14	10	9	1.01
2	636	21.08	20.86	1.05	22	21	5.31
3	981	23.77	23.77	9.71E−14	24	24	17.17
4	1700	33.23	33.23	1.87E−13	34	34	66.23
5	2111	46.5	45.71	1.69	47	46	146.47
6	3229	57.52	57.21	0.53	58	58	429.56
7	3852	76.51	76.51	1.83E−13	77	77	743.093
8	5160	84.56	84.56	2.93E−13	85	85	1484.66
9	7061	106.59	105.21	1.3	107	106	3395.15
10	8794	124.33	124.33	6.00E−13	125	125	6061.11

4 Conclusions

This paper presents different strategies to implement a column generation method for problem solving in delineating rectangular management zones in agricultural fields. More precisely, four strategies were proposed for selecting an efficient implementation of a column generation algorithm for solving a linear relaxation of the considered problem. These strategies combine some heuristics for solving the pricing problem and different stopping criteria through dual based lower bound for early finishing of the algorithm.

Numerical examples with sets of sample points of different sizes were carried out, which show that the strategy of adding more than one column to the new reduced Master Problem in each iteration leads to better results both in the quality of generated bounds for the optimal value as well as in computational time. No differences were observed in the quality of solutions obtained for integer variables, so that the results also show the quality of the adopted methodology.

Acknowledgements This research was partially supported by DGIP (Grant USM 28.13.69 and Graduate Scholarship), and CIDIEN of the Universidad Técnica Federico Santa María.

References

1. Barnhart, C., Johnson, E.L., Nemhauser, G.L., Savelsbergh, M.W.P., Vance, P.H.: Branch-and-price: column generation for solving huge integer programs. Oper. Res. **46**(3), 316–329 (1998)
2. Cid-García, N.M., Albornoz, V., Ríos-Solís, Y.A, Ortega, R.: Rectangular shape management zone delineation using integer linear programming. Comput. Electron. Agric. **93**, 1–9 (2013)
3. Desaulniers, G., Desrosiers, J., Solomon, M.M.: Column Generation. Springer, New York (2005)
4. Ghoniem, A., Sherali, H.: Complementary column generation and bounding approaches for set partitioning formulations. Optim. Lett. **3**(1), 123–126 (2009)
5. Nemhauser, G.: Column generation for linear and integer programming. In: Extra Volume: Optimization Stories, Documenta Mathematica, pp. 65–73. Fakultät für Mathematik, Universität Bielefeld, Germany (2012)
6. Ortega, R., Santibáñez, O.A.: Determination of management zones in corn (zea mays l.) based on soil fertility. Comput. Electron. Agric. **58**, 49–59 (2007)
7. Rönnberg, E., Torbjörn, L.: All-integer column generation for set partitioning: basic principles and extensions. Eur. J. Oper. Res. **233**(3), 529–538 (2013)
8. Vanderbeck, F., Wolseym L.A.: An exact algorithm for ip column generation. Oper. Res. Lett. **19**(4), 151–159 (1996)

Modeling Multi-Stage Decision Optimization Problems

Ronald Hochreiter

1 Introduction

We consider a multi-stage stochastic decision optimization framework based on a discrete-time decision process, i.e. there is a sequence of decisions at decision stages $t = 0, \ldots, T$ where at each stage t a decision taker observes the realization of a random variable ξ_t, and takes a decision x_t based on all observed values $\xi_0, \ldots, \xi_t$. At the terminal stage T a sequence of decisions $x = (x_0, \ldots, x_T)$ with respective realizations $\xi = (\xi_0, \ldots, \xi_T)$ leads to some cost $f(x, \xi)$. The goal is to find a sequence of decisions $x(\xi)$, which minimizes a functional of the cost $f(x(\xi), \xi)$. Multi-stage means that there is at least one intermediary stage between root stage and terminal stage.

The design goal of the approach presented in this paper is to design a modeling language independent of (a) the optimization modeling approach, e.g. expectation-based convex multi-stage stochastic programming or worst-case optimization, as well as (b) the underlying solution technique, e.g. either solving a scenario tree-based deterministic equivalent formulation or computing upper and lower bounds using primal and dual decision rules. Finally the modeling language should (c) be completely independent from a concrete programming language (C/C++, R, MatLab, Python, …). The idea is to compose a meta model and instance concrete implementations semi-automatically.

Consider the two most common ways to solve multi-stage decision optimization problems, which is on one hand the scenario-based three-layered approach as shown in Fig. 1. See [9] for an overview of the area of stochastic programming, and

R. Hochreiter (✉)
Department of Finance, Accounting and Statistics, WU Vienna University of Economics and Business, Vienna, Austria
e-mail: ronald.hochreiter@wu.ac.at

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_27

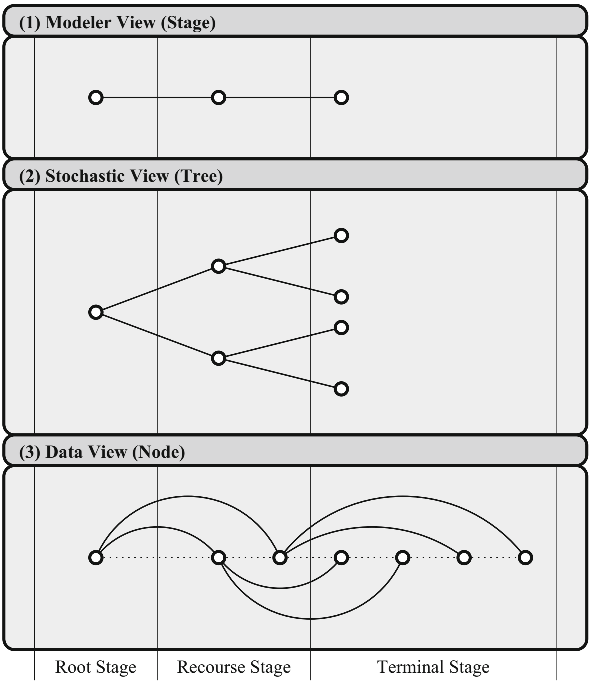


Fig. 1 Scenario tree-based three-layered approach

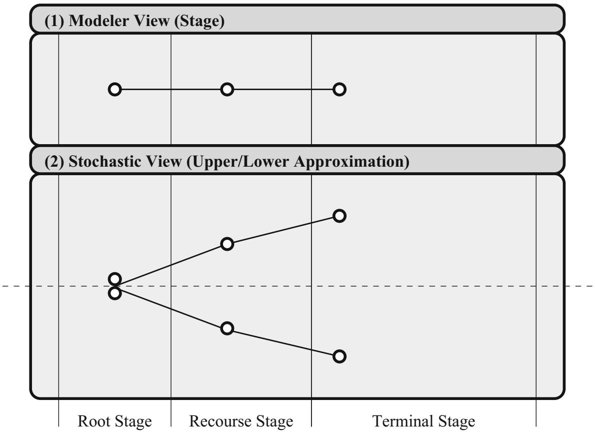


Fig. 2 Scenario tree-free approximation

[13] for stochastic programming languages, environments, and applications. More information on the modeling aspect can be found in [4].

The same decision problem may also be solved using scenario tree-free approximations [6] as shown in Fig. 2. The modeling language should be flexible enough to allow for applying any solution method, i.e. not being based on scenario trees, which is what most modeling language extensions for multi-stage models are proposing, see e.g. [1, 7, 11, 12], and [10].

2 Multi-Stage Convex Stochastic Programming

Consider a multi-variate, multi-stage stochastic process ξ and a constraint-set $\mathscr{X}$ defining a set of feasible combinations (x, ξ). The set $\mathscr{N}$ of functions $\xi \mapsto x$ are such that x_t is based on realizations up to stage t, i.e. only $(\xi_0, \ldots, \xi_t)$. These are the non-anticipativity constraints. This leads to the general formulation shown in Eq. (1).

$$\begin{array}{ll} \text{minimize}\, x : & \mathbb{F}\big(f(x(\xi), \xi)\big) \\ \text{subject to} & (x(\xi), \xi) \in \mathscr{X} \\ & x \in \mathscr{N} \end{array} \tag{1}$$

The most common way to solve such a problem is to create a scenario tree approximation of the underlying stochastic process and to build a deterministic equivalent formulation. The problem is that most modeling environments and languages are solely focussing on this type and mostly provide linear-only models due to solvability concerns. Furthermore, most allow for text-book applications only. There is almost no flexibility provided to extend models to use real-world objective functions and constraints.

The proposed solution is based on a complete decoupling of any scenario tree type of modeling from the decision problem modeling process, as shown in Fig. 1. On the decision problem (modeling) layer one should only be concerned with actions and decisions at stages. Other layers differ depending on the chosen solution method. In case of scenario trees and deterministic equivalent formulations there is an explicit decoupling of modeling and (scenario) tree handling, i.e. a scenario tree layer, whose focus is to create a scenario tree which optimally represents the subjective beliefs of the decision taker at each node. Furthermore there is an additional data layer, which handles the way how to (memory-)optimally store large scenario trees, access ancestor tree nodes quickly, and other computational (tree) operations.

3 Multi-Stage Modeling Example

Consider the stylized simple multi-stage stochastic programming example from [3], which is shown in Eq. (2).

$$
\begin{aligned}
\text{minimize } & \mathbb{E}\big(\textstyle\sum_{t=1}^{T} V_t x_t \big) \\
\text{subject to } & s_t - s_{t-1} = x_t \quad \forall t = 2, \dots, T \\
& s_1 = 0, s_T = a, \\
& x_t \geq 0, s_t \geq 0.
\end{aligned}
\tag{2}
$$

The decision to be computed with this model is the optimal purchase over time under cost uncertainty, where the uncertain prices are given by V_t, and the decisions x_t are amounts to be purchased at each time period t. The objective function aims at minimizing expected costs such that a prescribed amount a is achieved at T; s_t is a state variable containing the amount held at time t.

In Table 1 a concise meta formulation of this problem can be seen. The general syntax is borrowed from algebraic modeling languages like AMPL [2] and ZIMPL [5].

The most striking feature is that any relation to stages is removed from the definition of the optimization model—parameters, variables, objective function, and constraints. To accommodate for the definition of stages, the proposed stochastic modeling language contains two additional keywords for any of these objects, i.e.

- `deterministic` *`objects`*: *stage-set*;
- `stochastic` *`objects`*: *stage-set*;

Speaking in scenario tree notation the stochastic objects are defined on the underlying node structure and deterministic objects are defined on the stage

Table 1 Modeling formulation of Eq. (2)

```
deterministic a: T;
stochastic x, s, objective_function: 0..T;
stochastic non_anticitpativity: 1..T;
stochastic root_stage: 0;
stochastic terminal_stage: T;

param a;
var x >= 0, s >= 0;

minimize objective_function: E(V * x);
subject to non_anticitpativity: s - s(-1) = x;
subject to root_stage: s = 0;
subject to terminal_stage: s = a;
```

Table 2 Implementation of model 1 using the language R

```
m <- model()
parameter(m, a)
variable(m, x, lb=0)
variable(m, s, lb=0)

minimize(m, "objective", "E(V * x)")
subject_to(m, "non_anticitpativity", "s - s(-1) = x")
subject_to(m, "root_stage", "s = 0")
subject_to(m, "terminal_stage", "s = a")

deterministic(m, "T", a)
stochastic(m, "0..T", x, s, "objective")
stochastic(m, "1..T", "non_anticipativity")
stochastic(m, "0", "root_stage")
stochastic(m, "T", "terminal_stage")

optimize(m)
```

structure, i.e. the latter contain the same value for all nodes in the respective stage. To define stochastic objective functions and stage recourse the following functions are defined, e.g. the most commonly used expectation functional for objective functions is simply expressed by the function `E()`. Furthermore, there is a special way to define stage-wise recourse for stochastic variables, i.e. `variable-name(recourse-depth)`. Note that while most modeling languages allow for a single stage recourse only, this definition allows for any number of recourse stages.

Table 2 shows the modeling example in some concrete implementation for the statistical computing language R [8]. This definition can be easily converted to a deterministic equivalent formulation or any other reformulation—all information is available in a concise format.

4 Conclusion

In this paper, a modeling language framework for a successful simplified meta modeling of multi-stage decision problems under uncertainty is shown, which allows for automatic reformulation and solution of multi-stage problems. This can be seen as a basis to build a model-based multi-stage problem library, especially because of its inherent decoupling from the underlying optimization technique as well as the fact that it is not bound to a specific programming language. Furthermore it is easy to integrate robust and stochastic optimization techniques to allow for comparing solutions to determine, which approach is optimally suited for which class of decision models. There are many ways to extend the proposed meta language—possible straight-forward extensions are e.g. quantiles for objective functions. In addition, application-related risk measures (shortcuts) can be defined, e.g. `CVaR(objects)`, as well as probabilistic constraints.

References

1. Colombo, M., Grothey, A., Hogg, J., Woodsend, K., Gondzio, J.: A structure-conveying modelling language for mathematical and stochastic programming. Math. Program. Comput. **1**(4), 223–247 (2009)
2. Fourer, R., Gay, D., Kernighan, B.: AMPL: A Modeling Language for Mathematical Programming, 2nd edn. Duxbury/Brooks/Cole Publishing Company, Boston (2002)
3. Heitsch, H., Römisch, W., Strugarek, C.: Stability of multistage stochastic programs. SIAM J. Optim. **17**(2), 511–525 (2006)
4. King, A., Wallace, S.: Modeling with Stochastic Programming. Springer Series in Operations Research and Financial Engineering. Springer, New York (2013)
5. Koch, T.: Rapid mathematical prototyping. Ph.D. thesis, Technische Universität Berlin (2004)
6. Kuhn, D., Wiesemann, W., Georghiou, A.: Primal and dual linear decision rules in stochastic and robust optimization. Math. Program. **130**(1), 177–209 (2011)
7. Lopes, L.B.: Modeling stochastic optimization: from idea to instance. Ph.D. thesis, Northwestern University (2003)
8. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna (2013). http://www.R-project.org
9. Ruszczyński, A., Shapiro, A. (eds.): Stochastic Programming. Handbooks in Operations Research and Management Science, vol. 10. Elsevier, Amsterdam (2003)
10. Thénié, J., Delft, C.V., Vial, J.P.: Automatic formulation of stochastic programs via an algebraic modeling language. Comput. Manag. Sci. **4**(1), 17–40 (2007)
11. Valente, C., Mitra, G., Sadki, M., Fourer, R.: Extending algebraic modelling languages for stochastic programming. INFORMS J. Comput. **21**(1), 107–122 (2009)
12. van Delft, C., Vial, J.P.: A practical implementation of stochastic programming: an application to the evaluation of option contracts in supply chains. Automatica **40**(5), 743–756 (2004)
13. Wallace, S.W., Ziemba, W.T. (eds.): Applications of stochastic programming. MPS/SIAM Series on Optimization, vol. 5. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2005)

A New and Innovative Approach to Assess and Quantify the Value for the Customer

Susana Nicola, Eduarda Pinto Ferreira, and J.J. Pinto Ferreira

1 Introduction

Value for the Customer [21] is one of the most important factors in the success of an organization, maybe it is the master key to overcome the boundless challenges of this global competitive market. Moreover it is "the right goal for firms that seek to maximize long-term profits" [10]. "Satisfying customers is the source of a sustainable value creation" [5]. But the value derived by one individual is likely to be different from the value derived by another. So we can say, not only does each of us value the same things differently, we, individually, value different things at different times in different ways [21]. Value is a slippery concept which is very dependent on perception. To overcome this challenge we have to understand how customers assess and perceived the actual product/service.

Over many years some work have been made and discussed in the literature on the concept on Value for the Customer. Zeithmal has suggested customer perceived value as "what they get benefits relative to what they have to give up" (cost or sacrifices) [22]. Lai has suggested a framework for customer value focuses on the buyer's evaluation of product purchase at the time of buying, integrating cultural value, personal values, consumption values and product benefits [16]. Huber believed that benefits and costs are defined in terms of consumer's perceptions in the

S. Nicola (✉) • E.P. Ferreira
Porto Polytechnic-School of Engineering, GECAD, Rua Dr. Antonio Bernardino de Almeida, 431 Porto, Portugal
e-mail: sca@isep.ipp.pt; epf@isep.ipp.pt

J.J. Pinto Ferreira
Universidade do Porto, Faculdade de Engenharia/DEIG, INESC TEC/UITT, Rua Dr. Roberto Frias, Porto, Portugal
e-mail: jjpf@fe.up.pt

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_28

activities of acquisition, consumption and maintenance [14]. Flint creates a model to describe how customers' perceptions of value change over time in industrial supply relationship. This model centres in three forms of value: values, desired value and value judgement [12]. Woodruff defines customer value as "a customer's perceived preference for and evaluation of those product attributes, attribute performances and consequences arising from use that facilitate achieving the customer's goals and purposes in use situations" [12]. The authors Kothandaraman and Wilson[15] had developed a model based on three concepts of value creation: superior customer value, core capabilities and relationship. Ulaga developed a model for buyer-seller relationship and integrate the relationship value into the network relationship marketing [19].

This research proposes a modelling framework, the so-called "Conceptual Model Decomposing Value for the Customer" (CMDVC) and a quantitative model that aims at enabling the supplier enterprise to better understand how customer's perceive value and what are the key points to innovate or renovate the enterprise business to offer the customer an enhanced value proposition. Many studies had been explored in the "Value for the Customer" [21] and "Perceived Value" [18], but the decomposition of value is not dissected and broken down into its components, namely the firm assets used and build in the construction of the exchanged value, whether internal or external to the company or organization.

1.1 Value Temporal Positions

To understand how customers determine/perceive value in a sequential activity of a value proposition, Woodall [21] divides Value for the Customer into four value temporal positions: (a) pre-purchase—a phase of trying to predict how people perceive their services [14]; (b) at the point of trade—which implies a sense of VC experienced at the point of trade.; e.g. Acquisition Value plus Exchange Value; (c) post-purchase—a phase that delivers results of experiments based on customers'/suppliers' choices; e.g: use value; Received Value [14]; (d) after/use experience—a phase that reflects the point of disposal/sale.

1.2 Forms of Value

Furthermore, but linked to the above, Woodall classified Value for the Customer into different forms of value: (a) Net VC—"balance of benefits and sacrifices" to provide the best or the worst VC; (b) Marketing VC—"perceived the products attributes"; (c) Sale VC—primarily concerned with the price; (d) Rational VC—"difference from the objective price"; (e) Derived VC—users' experiences.

1.3 Collaborative Networks-ARCON

The Reference Model for Collaborative Network Organizations (ARCON) [7] provides a generic and abstract representation that enables the understanding of all involved entities and the relationships between all of them. We focus on the enterprise endogenous and exogenous assets under the following four dimensions: Structural (ST), Componential (CP), Functional (FUNC) and Behavioural (BEH). On the other hand, the outside perspective is captured by the exogenous elements that reveal the interaction with the surrounding environment and are divided into four dimensions: Market (MARK), Support (SUP), Societal (SOC), and Constituency (CONS).

This paper discusses the application of the Conceptual Model Decomposing Value for the Customer framework and a quantitative model used to access the adequacy of both enterprise offering to the customer needs and of its supporting assets.

2 Methodology

This research project followed the "Design Science" approach proposed by Hevner, March et al. "through the building and evaluation of artifacts designed to meet the identified business need" [13]. This approach provided the adequate setting for what we had in mind. In fact, we wanted to develop a new model (artifact), a "Conceptual Model Decomposing Value for the Customer" as well as an underlying quantitative model. In this sense, we extended and improved the "existing foundations in the design-science knowledge base" [13] in new and innovative way. The point we want to make is to determine how well our model work, using "information from the knowledge base (e.g. relevant research) to build a convincing argument for the artifact's utility" [13]. The research validation combined the Case Study Approach as described by Dube and Pare [11] and "informed arguments" from the literature review that helped build the case of the results validation [13]. Building on an Exploratory Case Study, and following the "design criterion in exploratory case research" [11], we first validated the proposed Conceptual Model for Decomposing Value for the Customer. It was in this context of looking at both the literature review and the business environment that the following research questions were tuned and designed:

1. *How can the Value for the Customer be modelled?*
   a. *How is this value built on top of assets endogenous and exogenous to the organization?*
   b. *How do endogenous and exogenous assets influence the Value for the Customer?*
2. *Can we derive a formal mathematical model that provides for the quantitative handling of the proposed model?*

3 Conceptual Model Decomposing Value for the Customer (CMDVC)

The proposed Conceptual Model for Decomposing Value for the Customer builds on a combination of the following concepts: (a) forms of value and Value temporal positions [21]; (b) Value Network exchanged tangible and intangible deliverables, building on the enterprise tangible and intangible assets [1–4]; and (c) enterprise Endogenous and Exogenous assets, concept extracted from ARCON, A Reference Model for Collaborative Network Organizations [6, 7].

This model comprises the understanding that time has direct impact in customer perceived value, because perceptions change from the pre-purchase phase to the post-purchase phase. We wanted to understand how value for the customer could be broken down into simpler constituents, integrating the value perceived by the enterprise members for a particular time position. The construction of the enterprise Value Network (through an interview with enterprise members), provides the identification of each deliverable (DL) exchanged with the customer, as well as the assets (endogenous and exogenous) built and/or used in the provision of that deliverable. This analysis further relates each deliverable (DL) with the forms of value. We apply the concepts proposed by the Reference Model for Collaborative Organizations, to classify the assets built and/or used as endogenous or exogenous to the enterprise. Figure 1 picture the three step approach to decomposing and assessing the value for the customer, enabling the systematic application of the process for future projects. In the 1st step of this process, on the right side of Fig. 1, we have the enterprise member's perspective. This shows: (1) how does the people inside the enterprise perceive the relative relevance of the assets involved in the process; and (2) how these assets relate to the Perceived Benefits (PBi)/Sacrifices (PSi). These two components are modelled as comparison matrices of the triangular fuzzy numbers resulting from: (1) each enterprise member assesses each asset relative relevance; and (2) assesses the relevance of each asset to each Perceived Benefit (PBi)/Sacrifice (PSi). The combination of these comparison matrices provide the input to a process that leads to the construction of the final matrix where we will be able to extract the most relevant assets and PBi/PSi. In the 2nd step of this process, the left side of Fig. 1, we have an extension of the Conceptual Model to enable an easier interaction with the customer (by reducing the burden of task demanded from the customer). In this context we try to obtain the further information from the enterprise client/customer for a particular Time Position and regarding his perception of benefits and sacrifices. The last step of the assessment of the enterprise Value Proposition and of its supporting assets was analysed combining the two described streams, the Enterprise perspective on the left and the Customer perspective on right. A computational implementation of the quantitative model was developed in PHP using MySQL database.

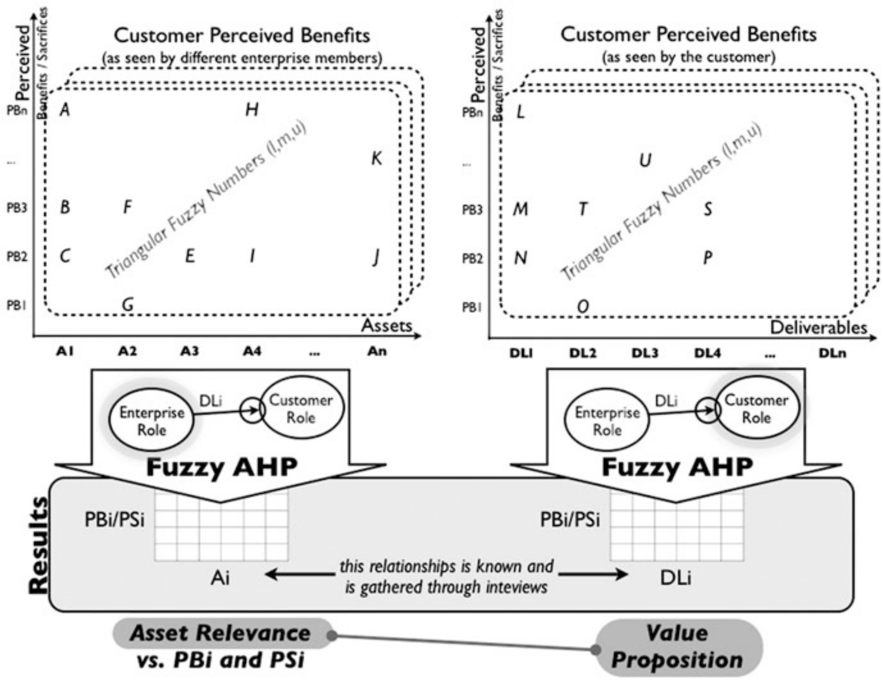


Fig. 1 Customer perceived value assessed by the enterprise members and enterprise customer for a particular time position

3.1 *The Method for Assessing and Integrating of both the Enterprise and the Customer Perspectives of the Perceived Value*

The Enterprise Perspective (1st Step) and the Customer Perspective (2nd Step)

For the enterprise and customer we have a several and conflicting criteria (Assets and Deliverables) and alternatives (Perceived Benefits/Sacrifices) where an assessment is not easily determined. The customer and the enterprise members will have to make their pair-wise comparison using the Saaty's scale for the deliverables and for the perceived benefits and sacrifices. The input information of both subjective judgements relating criteria and alternatives, is uncertain and imprecise. In this context, the fuzzy theory is usually applied to handle uncertain and subjective problems in the decision-making process. Therefore we apply the fuzzy Analytical Hierarchical Process (AHP) to solve this multi-criteria decision-making (MCDM) problem. The process unfolds as follows. Each enterprise member as well as the customer, performs an individual pair-wise comparison using the Saaty's scale.

Then a comprehensive pair-wise comparison matrix (Eq. (3)) is built by integrating the enterprise member's grades (b_{jep}) through Eqs. (1)–(2), [9], where enterprise members pair-wise comparison value is transformed into triangular fuzzy numbers.

$$l_{je} = min\left(b_{jep}\right), m_{je} = \frac{\sum_{p=1}^{t}\left(b_{jep}\right)}{p}, u_{je} = max\left(b_{jep}\right), \tag{1}$$

$$p = 1, 2, \cdots, t; \quad j = 1, 2, \cdots, m; \quad e = 1, 2, \cdots, m.$$

$$\tilde{b}_{je} = \left(l_{je};\ m_{je};\ u_{je}\right), \quad j = 1, 2, \cdots, m; \quad e = 1, 2, \cdots, m. \tag{2}$$

Then we apply the approach of Chang [8] for handling fuzzy AHP, by using the "extent analysis method" for the synthetic extent values, which derives crisp weights for fuzzy comparison matrix. Consider a triangular fuzzy comparison matrix Eq. (3) obtained by the steps of Chen [9]:

$$\tilde{D}_p = \left(\tilde{b}_{ij}\right)_{n\times n} = \begin{bmatrix} \tilde{b}_{11} & \tilde{b}_{12} & \cdots & \tilde{b}_{1m} \\ \tilde{b}_{21} & \tilde{b}_{22} & \cdots & \tilde{b}_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{b}_{m1} & \tilde{b}_{m2} & \cdots & \tilde{b}_{mm} \end{bmatrix}$$

$$= \begin{bmatrix} (1,1,1) & (l_{12}, m_{12}, u_{12}) & \cdots & (l_{1n}, m_{1n}, u_{1n}) \\ (l_{12}, m_{12}, u_{12}) & (1,1,1) & \cdots & (l_{2n}, m_{2n}, u_{2n}) \\ \vdots & \vdots & \ddots & \vdots \\ (l_{n1}, m_{n1}, u_{n1}) & (l_{n2}, m_{n2}, u_{n2}) & \cdots & (1,1,1) \end{bmatrix} \tag{3}$$

where $\tilde{b}_{ij} = \left(l_{ij}, m_{ij}, u_{ij}\right) = \tilde{b}_{ij}^{-1} = \left(\frac{1}{u_{ij}}, \frac{1}{m_{ij}}, \frac{1}{l_{ij}}\right)$ for $i, j = 1, \cdots, n$ and $i \neq j$.

To calculate a priority vector of the above triangular fuzzy comparison matrix $\tilde{D}_p$, the steps of Chang's extent analysis can be given as in the following:

(1) First, sum up each row of the fuzzy comparison matrix $\tilde{D}_p$, by applying the fuzzy arithmetic operations:

$$\sum_{j=1}^{n} \tilde{b}_{ij} = \left(\sum_{j=1}^{n} l_{ij}, \sum_{j=1}^{n} m_{ij}, \sum_{j=1}^{n} u_{ij}\right), \quad i, j = 1, 2, \cdots, n. \tag{4}$$

Then the inverse of the vector (Eq. (5)) above is:

$$\left[\sum_{j=1}^{n} \tilde{b}_{ij}\right]^{-1} = \left(1/\sum_{j=1}^{n} u_{ij}, 1/\sum_{j=1}^{n} m_{ij}, 1/\sum_{j=1}^{n} l_{ij}\right). \tag{5}$$

(2) Second we normalize the rows sums (Eq. (6)) by:

$$\tilde{S}_i = \sum_{j=1}^{n} \tilde{b}_{ij} \times \left[\sum_{j=1}^{n} \tilde{b}_{ij} \right]^{-1}. \tag{6}$$

(3) Third, compute the degree of possibility for $\tilde{S}_i \geq \tilde{S}_j$ of two TFNs $\tilde{S}_i = (l_i, m_i, u_i)$ and $\tilde{S}_j = \left(l_j, m_j, u_j\right)$ by the following Eq. (7):

$$V\left(S_i \geq S_j\right) = \begin{cases} 1, & if\ m_i \geq m_j \\ 0, & if\ l_j \geq u_i \\ \frac{l_j - u_i}{(m_i - u_i) - \left(m_j - l_j\right)}, & otherwise \end{cases} \tag{7}$$

(a) In general, the priority weights are calculated by using Eq. (8):

$$d'\left(A_i\right) = min\ V\left(S_i \geq S_k\right) \quad k = 1, 2, \cdots, n; k \neq i. \tag{8}$$

are the pair wise comparison of the $\tilde{S}$ TFNs.

(b) Then the weight vector is given by Eq. (9):

$$W' = \left(d'\left(A_1\right); d'\left(A_2\right); \cdots; d'\left(A_n\right)\right)^T. \tag{9}$$

(c) Finally we normalized the weight vector (Eq. (10))

$$W = \left(d\left(A_1\right); d\left(A_2\right); \cdots; d\left(A_n\right)\right)^T. \tag{10}$$

where $W$is a non-fuzzy number.

By applying the fuzzy AHP method we obtain a matrix of overall results of the enterprise member perception of the relevant assets and the relevant PBi/PSi.

Integrating the Two Perspectives (3rd Step)

With these two matrixes we have the degree of priority one criterion or alternative against all others in a fuzzy comparison matrix, [20]. On the left we have the degree of priority (relevance) as seen by the enterprise of an Asset and its relation to a PBi/PSi, whereas on the right we have the degree of priority (relevance) as seen by the customer of deliverable and its relation to a PBi/PSi. The relationship between the assets and the deliverables is known, which means that one now should be able to understand how the enterprise assets (endogenous or exogenous) relate to PBi/PSi, thus enabling the tuning of the enterprise offer Value Proposition.

4 Conclusion

The Conceptual Model for Decomposing Value for the Customer (CMDVC) is a novel framework for modelling value and a useful tool for enterprises to better understand how value is perceived by their customers' in the context of the value proposition. The developed tool builds on a mathematical formulation for the CMDVC as well as on a computational implementation. We envisage the possibility of using this tool to assess perceived value of a particular offer and of redesigning the actual product/service offer to better meet the customers' needs through the preparation of a new proposal.

The proposed quantitative model revealed its usefulness by providing the discovery of previously disregarded connections between assets used and/or built in the foreseen exchange of deliverables and perceived benefits. In general, people of the enterprise would likely realize that some of their expectations regarding the customer perceived value may not be what they think and that adjustments are needed. This was evident in the three case studies and from the comments we had of enterprise members and customer interviews

> (. . .) this novel approach can be quite useful for CPMT to better manage its service offering and marketing approach (1st case study) [17]
> (. . .) looking at these results, it is very interesting to note what customers value and their perceptions of certain deliverables (2nd case study)
> (. . .) the model and the quantitative method becomes useful for the company, we had never realize how the technical competence was linked with the Dl5 and DL4 (3rd case study).

As main benefits of this research, we would highlight that this tool may be useful to help these companies in the generation of an internal discussion of how their offer is perceived by their clients. In all case studies it was interesting to realize that some unexpected variables emerged as being more relevant that initially thought. From the management perspective this brought up the awareness on those issues that may now be looked upon in a new way.

We thus hope that our research on Value for the Customer as well as concerning the development of the CMDVC will contribute not only to extend and improve the existence knowledge foundations. We further hope produce significant value to the enterprises, building the bridge between the Value for the Customer, customer perception of value, and the enterprise endogenous and exogenous assets, whether it is applied to negotiation setting or in the context of the value proposition. As a future research, we would like to extend our research developing a set of case studies to perform this study for different value temporal positions, namely at the point of the trade, in a post-purchase phase and after use experience.

References

1. Allee, V.: Reconfiguring the value network. J. Bus. Strategy **21**, 36–39 (2000)
2. Allee, V.: The value evolution: addressing larger implications of an intellectual capital and intangibles perspective. J. Intellect. Cap. **1**, 17–32 (2000)
3. Allee, V.: A value network approach for modeling and measuring intangibles. A paper presented at Transparent Enterprise, Madrid (2002)
4. Allee, V.: Value network analysis and value conversion of tangible and intangible assets. J. Intellect. Cap. **9**(1), 5–24 (2008). doi:http://dx.doi.org/10.1108/14691930810845777
5. Barnes, C., Blake, H., Pinder, D.: Creating and Delivering Your Value Proposition, Managing Customer Experience for Profit. Kogan Page, London (2009)
6. Camarinha-Matos, L., Afasarmanesh, H.: Arcon reference models for collaborative networks. In: Collaborative Networks: Reference Modeling, pp. 83–112. Springer Science+Business Media LCC, New York (2008)
7. Camarinha-Matos, L., Afasarmanesh, H.: Collaborative Networks: Reference Modeling. Springer Science+Business Media LCC, New York (2008)
8. Chang, D.Y.: Applications of the extent analysis method on fuzzy ahp. Eur. J. Oper. Res. **95**, 649–655 (1996)
9. Chen, H.: A research based on fuzzy ahp for multi-criteria supplier selection in supply chain. Master's Thesis, National University of Science and Technology, Taiwan (2005)
10. Day, G.S., Moorman, C.: Strategy from the Outside Profit from Customer Value. McGraw-Hill, New York (2010)
11. Dube, L., Pare, G.: Rigor in information systems positivist case research: current practices, trends, and recommendations. MIS Q. **27**(4), 597–636 (2003)
12. Flint, D.J., Woodruff, R.B., Gardial, S.F.: Customer value change in industrial marketing relationships—a call for new strategies and research. Ind. Mark. Manag. **26**, 163–175 (1997)
13. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design science in information systems research. MIS Q. **28**(1), 75–105 (2004)
14. Huber, J., Lynch, J., Al, E.: Thinking about values in prospect and retrospect: maximising experience utility. Mark. Lett. **8**, 323–334 (1997)
15. Kothandaraman, P., Wilson, D.T.: The future of competition: value-creating networks. Ind. Mark. Manag. **30**(4), 379–389 (2001)
16. Lai, A.W.: Consumer values and product benefits and customer value: a consumption behavior approach. Adv. Consum. Res. **22**, 381–388 (1995)
17. Nicola, S., Pinto Ferreira, E., Pinto Ferreira, J.: A novel framework for modelling value for the customer, an essay on negotiation. Int. J. Inf. Technol. Decis. Mak. **11**(3), 661–703 (2012). doi:10.1142/S0219622012500162
18. Sanchez-Fernandez, R., Iniesta-Bonillo, M.: The concept of perceived value: a systematic review of the research. Mark. Theory **7**, 427–451 (2007)
19. Ulaga, W., Egger, A.: Relationship value and relationship quality: broadening the nomological network of business-to-business relationships. Eur. J. Mark. **40**(3–4), 311–327 (2006)
20. Wang, Y.M., Luo, Y., Hua, Z.: On the extent analysis method for fuzzy ahp and its applications. Eur. J. Oper. Res. **186**, 735–747 (2008)
21. Woodall, T.: Conceptualising value for the customer: an attributional, structural and dispositional analysis. Acad. Mark. Sci. Rev. **12**, 1–42 (2003)
22. Zeithaml, V.A.: Consumer perceptions of price, quality, and value: a means-end model and synthesis of evidence. J. Mark. **52**, 2–22 (1988)

Bus Driver Rostering by Column Generation Metaheuristics

Vítor Barbosa, Filipe Alvelos, and Ana Respício

1 Introduction

Personnel scheduling or rostering [7] consists in defining the schedule of work for each of the workers in a company for a given period, the rostering period. A roster defines the schedules for all workers during the rostering period. A schedule for a single worker defines, for each day, which tasks have to be performed. The days when no tasks are assigned, represent days off. Companies usually have diverse tasks to assign on each day, sometimes needing particular skills, which gives rise to the rostering problem. The rostering problem consists in assuring the assignment of all company duties, using the available workers and respecting the labour and company rules. Rostering problems arise in distinct types of business as surveyed in [7] and [10]. Rostering problems have been addressed by several methods, as also shown in [7] and [10].

In this paper, we address the monthly (28 days) bus driver rostering (BDR) problem proposed in [9]. The tasks to be performed by the drivers are sequences of

V. Barbosa (✉)
Escola Superior de Ciências Empreariais, Instituto Politécnico de Setúbal, 2914-503 Setúbal, Portugal
e-mail: vitor.barbosa@esce.ips.pt

F. Alvelos
Centro Algoritmi and Departamento de Produção e Sistemas, Universidade do Minho, Campus de Gualtar, 4710-057 Braga, Portugal
e-mail: falvelos@dps.uminho.pt

A. Respício
Centro de Matemática, Aplicações Fundamentais e Investigação Operacional/Departamento de Informática, FCUL, Universidade de Lisboa, 1749-016 Lisboa, Portugal
e-mail: respicio@di.fc.ul.pt

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_29

trips and rest periods, and are previously defined as tasks. There is a fixed number of available drivers, but each is only considered as used if at least one task is assigned to him, incurring in a fixed cost to the company. Each task from each day of the rostering period must be assigned to a driver, to assure the company' service, however the schedule of each driver must comply with the labour rules to assure minimum rest-time between tasks, minimum number of days-off by week and on Sundays, maximum allowed work time by week and by month. These rules prevent the assignment of an early task if the driver had work in a late task in the previous day. In the beginning of the rostering period, information from the previous roster is considered to know the drivers who worked in the late tasks of the last day, preventing the assignment of early tasks to them, and the number of work days since the last day-off, to know when to assign the next day-off, according to the maximum number of consecutive days without a day-off. The objective is to build the roster minimizing the global cost, composed by the fixed cost paid by using a driver and the overtime cost, paid in the tasks with duration above the contractual workday duration.

We address the BDR problem by a SearchCol (short for "Metaheuristic search by column generation") algorithm. SearchCol [1, 2] is a recently proposed approach hybridizing column generation (CG) with metaheuristics (MHs). CG is a decomposition approach where linear programming problems and subproblems are iteratively solved in order to deal with linear/integer problems with a huge number of variables (the interested reader is referred to [6] for a detailed description and applications of CG). CG allows solving the linear relaxation of a integer programming model.

Unlike in branch-and-price (combination of CG and branch-and-bound) algorithms, in a SearchCol algorithm, the solution space is searched by MHs based on representing a (global) solution as made of one solution from each subproblem. SearchCol is a general framework which can address distinct problems provided that a decomposition model solvable by CG and a MH based on the aforementioned solution representation exist.

The decomposition model used in the proposed approach was first introduced in [4] and can be seen as a Dantzig-Wolfe decomposition [5] of the compact model proposed by Moz et al. in [9]. The proposed SearchCol algorithm has components similar to the ones introduced in [3, 4], where the CG stage of the algorithm is run only once, and then the search proceeds over the produced columns. In this version, two major differences arise: a short time limit is imposed for CG and the perturbations are introduced, allowing multiple iterations of the SearchCol cycle which includes the CG, with additional constraints, followed by the search.

This paper is organized as follows. Next section introduces a decomposition model for the BDR problem and presents how its linear relaxation is solved using CG. In Sect. 3, the main components of the proposed algorithm are presented, with a generic overview of a SearchCol algorithm, followed by a description of the evolutionary algorithm (EA) and also a description of the perturbations used. Section 4 describes the computational tests and present the obtained results. The paper ends with some conclusions.

2 Column Generation for Bus Driver Rostering

The decomposition model proposed in [4] is based on the definition of the decision variables $\lambda_j^v = 1$ if schedule j is the selected schedule for driver v and $\lambda_j^v = 0$ otherwise. The set of all valid schedules for driver v is represented by J^v. The cost of the schedule j of driver v is represented by p_j^v. Parameter a_{ih}^{jv} is equal to 1 if task i of day h belongs to schedule j of driver v. V is the set of available drivers and T_h^w defines the sets of tasks to assign on day h. The decomposition model is:

$$Min \sum_{v\in V}\sum_{j\in J^v} p_j^v \lambda_j^v \tag{1}$$

Subject to:

$$\sum_{v\in V}\sum_{j\in J^v} a_{ih}^{jv}\lambda_j^v \quad \geq 1\ ,\ i\ \in\ T_h^w, h = 1,\ldots,28, \tag{2}$$

$$\sum_{j} \lambda_j^v = 1\ ,\ j\ \in\ J^v,\ v\ \in\ V, \tag{3}$$

$$\lambda_j^v \in \{0,1\}\ ,\ j\ \in\ J^v,\ v\ \in\ V. \tag{4}$$

The objective function (1) minimizes the sum of the costs of the schedules included in the solution. The set of linking constraints (2) assure the assignment of all task and the convexity constraints (3) assure the assignment of a schedule to each driver.

Since the enumeration of all possible schedules is intractable, CG is used to solve the linear relaxation of the decomposition model. In CG, a restricted version of the decomposition model is defined by considering a subset of the decision variables (the so called restricted master problem—RMP). The RMP is enlarged by solving subproblems (each one associated with a driver) which return feasible schedules with negative reduced costs.

The solutions of the subproblems are integer, however, the final solution obtained from CG may be fractional since a linear relaxation is being solved. The rationale behind the application of SearchCol to the BDR problem is that even if a valid roster is not obtained from CG, the schedules generated when solving the subproblems can be combined to build quality rosters. In the next section, after an overview of SearchCol, we explain how that search space can be explored by an EA to find valid rosters.

3 SearchCol

The two main concepts of the SearchCol framework are (1) the usage of metaheuristics based on solutions obtained by the subproblems of CG and (2) the usage of perturbed CG (i.e. CG with additional constraints defined in terms of the subproblem variables). Figure 1 presents an overview of the global SearchCol algorithm.

Fig. 1 SearchCol algorithm

```
1: Column generation
2: Metaheuristic search
3: repeat
4:     Define column generation perturbation
5:     Optimize perturbed column generation
6:     Metaheuristic search
7: until Stopping criterion fulfilled
```

In the first step, CG is used to obtain a solution to the linear relaxation of the decomposition model presented in Sect. 1. In step 2, the set of subproblem solutions generated by the CG algorithm, are used as components for global solutions which define the search space of a metaheuristic. The cycle of steps 3 to 7 repeat those two steps but with additional subproblem solutions which are generated when CG is applied with additional constraints on subproblem variables.

In the search steps (steps 2 and 6 of the algorithm) of a SearchCol algorithm a metaheuristic is used. In this paper, as discussed in the next two subsections, we further explore the EA proposed in [3, 4] and the use of perturbations.

3.1 The Evolutionary Algorithm Metaheuristic

Evolutionary Algorithms are based on the evolution of the biological species, as proposed in [8], and work over populations of individuals, each one represented by a chromosome which describes all the characteristics of the individual.

Following the SearchCol framework and the BDR problem decomposition used, an individual/chromosome represents a roster, each gene position identifies a driver, and the gene content identifies the schedule assigned to that driver.

A population is obtained by using solution generators, which are defined in the SearchCol framework, until the desired population size. Details about the available generators are presented in [2]. Each generator creates an individual selecting a schedule for each driver. The use of distinct generators promotes diversity in the population.

The evaluation of an individual considers two dimensions: the first one, called feasibility, measures the cost of the roster; the second, called infeasibility, measures the number of unassigned tasks. A roster is only considered valid if the number of unassigned tasks is zero, assuring the company service.

The selection operator is the binary tournament and the variation operators are the classical one point and two points crossovers and a mutation operator, which change the schedule for the selected subproblems/gene locus, by randomly selecting a new one from the pool of schedules generated in the CG for that driver. An elitism mechanism is included to assure the preservation of the best individuals through generations.

A local search heuristic is used to explore solutions near the best solution found in each iteration. Local search is based in the definition of a neighbour as a solution in which one and only one driver has a schedule different from the one in the current solution. The EA stops when a number of consecutive iterations without improvement in the best solution is achieved.

3.2 Perturbations

In the SearchCol framework, a perturbation consists in fixing subproblem variables to 0 or to 1 by adding constraints in the RMP. In the case of the BDR, the basic idea is to force the assignments of chosen tasks to chosen drivers. When adding a perturbation to the RMP forcing the assignment of a task i to a driver v, all schedules generated by the subproblem of driver v will include task j. In the BDR, the use of perturbation intends to help in the achievement of valid rosters when the first search does not find one. Given the solution found, the perturbation fixes the assigned tasks to the drivers and then a new CG and metaheuristic cycle iteration is run to generate new schedules with the missing tasks (not assigned) and to search again for a better roster. Even in the valid rosters, the use of perturbations can be useful since it is possible that, in an obtained roster, some tasks are over-assigned (to more than a driver) and the perturbations will fix the task to only one driver, forcing the generation of new schedules to the other driver(s) without that task. A detailed explanation of how the perturbations are used in the SearchCol and the defined perturbations are presented in [2]. We used the perturbations based in the incumbent, setting to 1 a proportion of 30 % of the subproblem variables with value 1 in the incumbent solution (chosen randomly).

4 Computational Tests

The algorithm was implemented in the computational framework SearchCol++ (http://searchcol.dps.uminho.pt), which allows a fast implementation of SearchCol algorithms.

The BDR problem test instances are the ones designated as P80 from [9] and [3, 4].

In the SearchCol framework configuration, the time limit of the initial CG was set to 30 s. In the beginning, the subproblems are solved using a heuristic. When no new attractive column are found, the heuristic is replaced by CPLEX to obtain optimal solutions. All subproblems are solved in each iteration and the corresponding attractive columns added. The number of main cycles (CG and search) of the SearchCol metaheuristic was set to 20, with 3 as the maximum number of iterations without improvement. The other CG runs (after the first) are limited to 10 s.

The population size of the EA contains 240 individuals (1/3 of individuals built picking totally at random subproblem solutions, 2/3 of individuals built picking random subproblem solutions with the probability of been selected biased by the

Table 1 SearchCol with EA results

	Time (s)		Feasibility			Δ Feasibility from [3]	
Instance	Average	Std dev.	Mean	Std dev.	Best	Average (%)	Best (%)
P80_1	130.16	9.03	3946.87	141.52	3819	−24.2	−17.4
P80_2	112.37	19.34	2924.80	93.87	2873	−27.4	−12.1
P80_3	158.51	61.56	5693.30	260.86	5297	−24.1	−17.2
P80_4	132.17	25.87	5183.43	434.30	4479	−12.9	−3.7
P80_5	128.01	24.80	3942.30	142.26	3813	−21.2	−13.7
P80_6	131.70	38.69	4620.27	327.64	4135	−12.8	−8.0
P80_7	136.55	33.55	5241.80	334.86	4547	−18.1	−15.7
P80_8	139.67	29.74	5984.67	209.95	5697	−17.0	−12.3
P80_9	135.02	35.27	5070.77	367.66	4308	0.4	−1.2
P80_10	141.46	21.10	5177.27	143.71	4953	−16.7	0.8

linear solution of CG, half biased by the first CG solution and half biased by the last CG solution). An elite population of 40 individuals and the local search configured in the first improvement found were used. Crossover and mutation probabilities are 80 and 20 %. The stopping criterion is 500 iterations without improvement.

For each instance, the metaheuristic was executed 30 times. Aggregated results are presented, for the 30 runs. In Table 1, columns under "Time (s)" display the mean and the standard deviation for the computational times in seconds; columns under "Feasibility" the solution values, namely, the mean, the standard deviation and the best value found for each instance in the 30 runs. The Δ columns show the difference on the average and best solution values compared with the results from [3]. Negative values identifies improvement. All the runs obtained a solution with all tasks assigned (infeasibility=0), which was not achieved in [3]. The results presented in Table 1 show a general improvement on the best results and also on the average values of all runs. Only the average value for instance P80_9 and the best value for instance P80_10 worsened.

5 Conclusions

In this paper we addressed a bus driver rostering problem where the tasks that composes the company service are assigned to drivers, respecting the labour rules and minimizing the cost of the drivers used and overtime paid. We proposed a SearchCol algorithm, combining CG, an EA, and perturbations based in the incumbent.

The obtained results were compared with the algorithm in [3] which didn't include perturbations, and considered a time limit for the CG far longer than the total running time used by the current algorithm. The algorithm here proposed clearly outperforms the previous one. The quality of the solutions is, in average, 10.1 %

better in terms of the best value of each instance and 17.4 % better on the average value of all runs.

Acknowledgements This work is financed by Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within projects "SearchCol: Metaheuristic search by column generation" (PTDC/EIAEIA/100645/2008) and PEst-OE/EEI/UI0319/2014.

References

1. Alvelos, F., Sousa, A., Santos, D.: Searchcol: metaheuristic search by column generation. In: Blesa, M., Blum, C., Raidl, G., Roli, A., Sampels, M. (eds.) Hybrid Metaheuristics. Lecture Notes in Computer Science, vol. 6373, pp. 190–205. Springer, Berlin/Heidelberg (2010)
2. Alvelos, F., Sousa, A., Santos, D.: Combining column generation and metaheuristics. In: Talbi, E. (ed.) Hybrid Metaheuristics, Studies in Computational Intelligence, vol. 434, pp. 285–334. Springer, Berlin/Heidelberg (2013)
3. Barbosa, V., Respício, A., Alvelos, F.: Genetic algorithms for the searchcol++ framework: application to drivers' rostering. In: Oliveira, J.F., Vaz, C.B., Pereira, A. (eds.) IO2013 - XVI Congresso da Associaç ao Portuguesa de Investigaç ao Operacional, pp. 38–47. Instituto Politécnico de Bragança (2013). http://hdl.handle.net/10198/8451
4. Barbosa, V., Respício, A., Alvelos, F.: A hybrid metaheuristic for the bus driver rostering problem. In: Vitoriano, B., Valente, F. (eds.) ICORES 2013 - Second International Conference on Operations Research and Enterprise Systems, pp. 32–42. SCITEPRESS, Barcelona (2013). doi:10.5220/0004283700320042
5. Dantzig, G.B., Wolfe, P.: Decomposition principle for linear programs. Oper. Res. **8**(1), 101–111 (1960)
6. Desaulniers, G., Desrosiers, J., Solomon, M.M.: Column Generation. Springer, New York (2005)
7. Ernst, A., Jiang, H., Krishnamoorthy, M., Owens, B., Sier, D.: An annotated bibliography of personnel scheduling and rostering. Ann. Oper. Res. **127**(1–4), 21–144 (2004)
8. Holland, J.H.: Adaptation in Natural and Artificial systems. MIT Press, Cambridge (1992)
9. Moz, M., Respício, A., Pato, M.V.: Bi-objective evolutionary heuristics for bus driver rostering. Public Transp. **1**(3), 189–210 (2009)
10. Van den Bergh, J., Belin, J., De Bruecker, P., Demeulemeester, E., De Boeck, L.: Personnel scheduling: a literature review. Eur. J. Oper. Res. **226**(3), 367–385 (2013)

A Matheuristic Based on Column Generation for Parallel Machine Scheduling with Sequence Dependent Setup Times

Filipe Alvelos, Manuel Lopes, and Henrique Lopes

1 Introduction

The importance of parallel machine scheduling problems study comes from both the theoretical and the practical perspectives. From the theoretical perspective, it is a generalization of the single machine problem and a particular case of problems arising in flexible manufacturing systems. From the practical perspective, it is important because we can find many examples of the use of parallel machines in the real world. For a literature review on parallel machines scheduling problems see, for example, [9]. The interest in scheduling problems involving setup times/costs exists for more 40 years. For a literature review on scheduling problems with setup times (as the one addressed in this paper) or costs see, for example [2]. The increasing research trend in the more recent decades is justified by the significant savings obtained when setup times/costs are explicitly considered for scheduling decisions in real world industrial/service problems [1].

Using the $\alpha|\beta|\gamma$ Graham classification [4], the problem addressed in this paper is classified as $R|a_k, r_j, s_{ij}|\sum w_j T_j$. This problem is strongly NP-hard, because its special case $P||\sum w_j T_j$ is known to be strongly NP-hard, even for a single

F. Alvelos (✉)
Centro Algoritmi and Departamento de Produção e Sistemas, Universidade do Minho, Campus de Gualtar, 4710-057 Braga, Portugal
e-mail: falvelos@dps.uminho.pt

M. Lopes
CIDEM-ISEP, School of Engineering, Polytechnic of Porto, 4200-072 Porto, Portugal
e-mail: mpl@isep.ipp.pt

H. Lopes
Centro Algoritmi, Universidade do Minho, Campus de Gualtar, 4710-057 Braga, Portugal

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_30

machine [6]. We use the acronym MS to denote the machine scheduling problem addressed in this paper.

Several approaches have been proposed for this and related problems. For a detailed literature review the interested reader is pointed to [7]. In that same reference a branch-and-price (B&P) algorithm was first presented providing optimal solutions for instances with up to 50 machines and 150 jobs.

The matheuristic (a mathematical programming based heuristic) proposed in this paper uses the same decomposition model (DM) and column generation (CG) algorithm of the B&P of [7]. As in B&P, the first step of the matheuristic (MH) is to solve the linear relaxation of the DM by CG. However, in the next step, instead of starting a search based on a (branch-and-bound) tree where the nodes are solved by CG, as B&P does, the MH uses the general purpose integer programming (GPIP) solver to obtain an integer solution to the last solved restricted master problem. Additional constraints are then introduced in CG to fix some of the original variables with the purpose of providing the GPIP solver with better columns. The procedure is repeated until a stopping criteria is met (e.g. a maximum number of iterations without improvement of the incumbent is reached).

The proposed MH is closely related with SearchCol, a framework for combining CG and metaheuristics [3]. The main difference is that in SearchCol, a metaheuristic is used instead of a GPIP solver in the search phase.

In the next section, the MS problem is formally defined and the DM is described. In Sect. 3 the MH is detailed. The performance of the MH is evaluated through comparative computational results in Sect. 4. The main conclusions are stated in Sect. 5.

2 Problem Definition and a Decomposition Model

In the MS problem there are n jobs to be scheduled in m machines. Each job $j, j = 1, \dots, n$, must be processed in exactly one machine. The machines are unrelated, i.e. the processing time of each job depends on the machine. The processing time of job j in machine k, $k = 1, \dots, m$, is represented by p_{jk}. Preemption of jobs is not allowed. A machine can only process one job at a time. There are setup times, dependent of the sequence. The setup time corresponding to a change from a job i to a job j is represented by s_{ij}. There is a release date, a due date, and weight associated with each job j represented by r_j, d_j, and w_j, respectively. Each machine k has an availability date, a_k, and at moment 0 is prepared (no setup required) for a job represented by l_k. The objective is to minimize the weighted tardiness which, representing the completion time of job j by C_j, can be represented by

$$\sum_{j=1}^{n} w_j Max(C_j - d_j, 0)$$

A decomposition model (DM) can be obtained by enumerating the set of all feasible schedules for all machines. We denote the set of all feasible schedules, indexed by p, of machine k by P^k. The number of times a job j is processed in schedule p of machine k is v_{jp}^k. Note that, for easiness of the solution of a subproblem to be described below, we allow a job to be processed more than one time. The cost of the schedule p of machine k is given by $c_p^k = \sum_{j \in J^{pk}} w_j Max(C_j - d_j, 0)$ where J^{pk} is the set of jobs in the schedule p of machine k (i.e. jobs with $v_{jp}^k \geq 1$).

We define decision variables as

$$y_p^k = \begin{cases} 1, & \text{if schedule } p \text{ of machine } k \text{ is selected for that machine} \\ 0 & \text{otherwise.} \end{cases}$$

The DM is:

$$Min \sum_{k=1}^{m} \sum_{p \in P^k} c_p^k y_p^k \tag{1}$$

$$subject\,to:$$

$$\sum_{p \in P^k} y_p^k = 1; k = 1, \ldots, m \tag{2}$$

$$\sum_{k=1}^{m} \sum_{p \in P^k} v_{jp}^k y_p^k \geq 1; j = 1, \ldots, n \tag{3}$$

$$y_p^k \in \{0, 1\}; k = 1, \ldots, m; p \in P^k$$

The objective function (1) minimizes the total weighted tardiness. Constraints (2) state that one schedule is chosen for each machine (possibly the empty schedule). Constraints (3) state that all jobs must be processed.

The DM is of practical use only if it can be addressed without explicitly considering all the decision variables—which are too many to be enumerated. CG allows solving the linear relaxation of the decomposition by considering a reduced set of decision variables which is enlarged iteratively by including variables associated with solutions of subproblems, each one associated with one machine.

The subproblem of machine k consists in identifying a variable with negative reduced cost, i.e. a schedule $p*$ for which $\bar{c_{p*}^k} < 0$ where

$$\bar{c_{p*}^k} = \sum_{j \in J^{p*k}} w_j Max(C_j - d_j, 0) - \pi_j$$

where π are the dual variables associated with constraints (3).

The subproblem of machine k can be modelled as a network where each elementary path between an initial node and a final node corresponds to a schedule. A node j is associated with job j and, if visited, offers a profit of π_j and consumes a time of p_j. An arc ij is associated with processing job j immediately after job i and consumes a time of s_{ij}. C_j corresponds to the sum of the times in the nodes and in the arcs visited before node j.

In [7], dynamic programming algorithms to solve the subproblems are described and used in the B&P algorithm (combination of CG with branch-and-bound). The states are defined by (j,t) where j is the last *job* processed and t is its completion time. Given that the states do not keep information on jobs processed before the last, this algorithm can produce cycles in the above mentioned network which correspond to process more than one time the same job(s) (that is why v_{jp}^k is a general integer and not binary). Significant improvements on this approach can be found in [8].

3 The Matheuristic

CG based approaches, in particular B&P, have been used to solve with success large instances of the problem addressed. We propose a heuristic based on CG and using the same concepts of SearchCol, metaheuristic search by column generation [3].

SearchCol has three main steps which are executed in cycle: (i) solve (perturbed) CG, (ii) search, (iii) update perturbations.

In step (i) perturbed CG refers to CG with additional constraints (perturbations) which will be discussed below. In the first iteration CG is solved with no perturbations, obtaining an optimal solution to the linear relaxation of the DM. In step (ii) the set of schedules obtained with column generation is used to perform a search for a global schedule (a set of schedules, one for each machine). In SearchCol that search is made through a metaheuristic (as VNS or tabu search), in the proposed MH the search corresponds to solve a restricted version of the DM (only the schedules generated by CG are included in the model) with a GPIP. Note that in step (i) a relaxation is solved but in step (ii) the integer (restricted) model is considered. A time limit is imposed in the GPIP (in the proposed MH was $0.1 \times (n+m)$ seconds).

CG provides the optimal solution of the linear relaxation of the DM. However, the optimal solution of the (not relaxed) DM may include schedules that were not generated by CG. With the purpose of conducting CG in generating desirable schedules, additional constraints (perturbations) are included in each CG step after the first iteration. Perturbations have many types and, as almost all components of SearchCol and therefore of the MH, can be described and implemented in a problem independent way. However, for briefness and clarity of the exposition, we only describe the ones used in the proposed MH (details can be found in [3]).

Perturbations are used to force subsets of jobs to be processed in certain machines. The first step when defining the perturbations to include in the CG of the next iteration is to sort by decreasing CG recency (i.e, the number of times a schedule was obtained as the solution of the CG subproblem) the schedules belonging to the incumbent. The first 10 % schedules are then used to define the perturbations of the current iteration. For each of those schedules, each one associated with a machine $\bar{k}$, the set of processed jobs is identified and, for each job,

$\bar{j}$, one constraint of the following form is included in the (restricted) master problem of CG:

$$\sum_{p \in P^{\bar{k}}} v_{\bar{j}p}^{\bar{k}} y_p^{\bar{k}} \geq 1.$$

In CG, the duals of the perturbation constraints are taken into account in the objective function of the subproblem.

In the following iteration, only machines that were not selected previously are considered in step (iii). When there are perturbations associated with all machines, all perturbations are removed and a restart happens. After one restart with no improvement of the incumbent or when a time limit is reached, the algorithm stops.

4 Computational Results

The MH was implemented and tested using SearchCol++ [10], an implementation of the SearchCol framework in C++. When using SearchCol++, only information on the decomposition and problem specific components must be coded. Steps (ii) and (iii) (search and perturbations) are hidden from the user and are controlled through input parameters.

SearchCol++ uses Cplex 12.4 [5] as the general purpose solver for the restricted master problems and also as the GPIP. The computational tests were performed on a notebook pc with a 2.4 GHz i3-3110M Intel processor and 4 Gb of RAM.

We compare our results with the branch-and-price algorithm of [7], both with a time limit of 1 h. We tested 10 sets with 10 randomly generated instances each. The instances in a set have the same $(m - n - q)$, where q is the congestion level which can take values 3 or 4 (the most difficult ones as shown in [7]). B&P obtained an optimal solution in 41 out of the 100 instances.

The average results for each set of 10 instances are presented in Table 1.

Table 1 can be read in two directions: reading left to right shows the impact of the increase of congestion, reading top to bottom shows the impact of the increase of the size of the instance.

Table 1 Differences (MH-B&P) between the times and values of the MH and B&P

Instances	Time	Value (%)	Better	Instances	Time	Value (%)	Better
50-200-3	677	3	0 (1)	50-200-4	−960	−9	2
50-220-3	1022	−8	2 (1)	50-220-4	−358	−22	6
60-240-3	411	3	3	60-240-4	−1016	−36	9
80-320-3	693	57	6	80-320-4	−534	−8	10
100-400-3	−505	−11	9	100-400-4	−228	−6	9

Negative values mean the result of the MH is better than the one of B&P. Column Better corresponds to the number of instances (out of 10) the MH was better than B&P (between parenthesis the number of solutions with the same value, if any). Time limits were set to 1 h. The first four columns are for instances with less congestion (level 3)

The proposed MH outperforms B&P for the instances with more congestion obtaining, in average, better solutions in less time. For the instances with less congestion, the proposed MH has better results than B&P for the set with larger instances.

The result of the MH for the set 80-320-3 is explained by the poor solutions obtained by the MH in three instances of that set. The MH provided better solutions by more than 10 % in 36 instances. The opposite happened in 7 instances.

5 Conclusions

In this paper, a matheuristic (MH) based on column generation (CG) and a general purpose integer programming (GPIP) solver was proposed to solve a parallel machine scheduling problem with sequence dependent setup times. The MH follows the SearchCol framework and has three steps which are run in a cycle: solve CG (after the first iteration with additional constraints—entitled perturbations), search the solution space provided by CG with the GPIP, update the perturbations.

The MH obtained better solutions in shorter times for very large instances when compared with a state-of-the-art branch-and-price algorithm.

Acknowledgements This work is financed by Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within projects "SearchCol: Metaheuristic search by column generation" (PTDC/EIAEIA/100645/2008) and PEst-OE/EEI/UI0319/2014.

References

1. Allahverdi, A., Soroush, H.M.: The significance of reducing setup times/setup costs. Eur. J. Oper. Res. **187**(3), 978–984 (2008)
2. Allahverdi, A., Ng, C., Cheng, T., Kovalyov, M.: A survey of scheduling problems with setup times or costs. Eur. J. Oper. Res. **187**(3), 985–1032 (2008)
3. Alvelos, F., Sousa, A., Santos, D.: Combining column generation and metaheuristics. In: Talbi, E.G. (ed.) Hybrid Metaheuristics. Studies in Computational Intelligence, vol. 434, pp. 285–334. Springer, Berlin/Heidelberg (2013)
4. Graham, R.L., Lawler, E.L., Lenstra, J.K., Kan, A.R.: Optimization and approximation in deterministic sequencing and scheduling: a survey. Ann. Discret. Math. **5**(2), 287–326 (1979)
5. ILOG: IBM ILOG CPLEX Optimization Studio V12.4 (2012)
6. Lenstra, J.K., Rinnooy Kan, A.H.G., Brucker, P.: Complexity of machine scheduling problems. Ann. Discret. Math. **1**, 343–362 (1977)
7. Lopes, M.J.P., de Carvalho, J.M.V.: A branch-and-price algorithm for scheduling parallel machines with sequence dependent setup times. Eur. J. Oper. Res. **176**(3), 1508–1527 (2007)
8. Lopes, M., Alvelos, F., Lopes, H.: Improving branch-and-price for parallel machine scheduling. Lect. Notes Comput. Sci. **8580**, 290–300 (2014)
9. Mokotoff, E.: Parallel machine scheduling problems: a survey. Asia-Pacific J. Oper. Res. **18**(2), 193–243 (2001)
10. SearchCol++: http://searchcol.dps.uminho.pt/ (2014)

Population Games with Vector Payoff and Approachability

Dario Bauso and Thomas W.L. Norman

1 Introduction

We consider a game played by a large population of individuals in continuous time. At every time, each individual faces a random opponent extracted from the population and the resulting payoff is a vector, which represents a collection of noninterchangeable goods.

As main result we provide a new model that combines approachability and population games. Given that the opponent is randomly extracted from the population, the approach by Blackwell—which looks at the worst-case payoff—may appear conservative. Thus, we relax Blackwell's conditions, assuming that the opponent is not malevolent but instead is simply extracted from a population with given distribution; we call this 1*st-moment approachability*.

We review next some related literature. The theory of "approachability" dates back to Blackwell [4] and culminates in the well-known Blackwell's Theorem. Approachability arises in several areas of game theory, such as allocation processes in coalitional games [17], regret minimization [9, 19], adaptive learning [5–8]. The original Blackwell's formulation of approachability has been extended to continuous-time repeated games, thus showing common elements with Lyapunov theory [9]. A definition of approachability in infinite-dimensional space has been provided by Lehrer [18]. A recent work of the first author [2] studies approachability

D. Bauso (✉)
DICGIM, Università di Palermo, Palermo, Italy
e-mail: dario.bauso@unipa.it

T.W.L. Norman
Magdalen College, Oxford, UK
e-mail: thomas.norman@magd.ox.ac.uk

R.J. Fonseca et al. (eds.), *Computational Management Science*, Lecture Notes in Economics and Mathematical Systems 682,
DOI 10.1007/978-3-319-20430-7_31

in an n-player repeated game. Approachability can be reframed within differential game theory [20, 21].

A second stream of literature we follow in the present study is the one on *mean-field games*. This theory originated in the work of Huang et al. [10–12], and independently in that of Lasry and Lions [14–16], where the now standard terminology of mean-field games was introduced. Mean-field games have connections to evolutionary games (see for instance [13]) and large games [1].

In the rest of this section we introduce notation. We view vectors as columns. For a vector x, we use x_i to denote its ith coordinate component. For two vectors x and y, we use $x < y$ ($x \leq y$) to denote $x_i < y_i$ ($x_i \leq y_i$) for all coordinate indices i. We let x^T denote the transpose of a vector x, and $\|x\|$ its Euclidean norm. We write $P_X(x)$ to denote the projection of a vector x on a set X, and $\mathrm{dist}(x,X)$ for the distance from x to X, i.e. $P_X(x) = \arg\min_{y\in X} \|x-y\|$ and $\mathrm{dist}(x,X) = \|x-P_X(x)\|$, respectively. We also denote by *conv* the convex hull of a given set of points. The symbol ∂_x indicates the first partial derivative with respect to x. Also, the symbol $\mathbb{E}$ denotes expectation and given a set of pure strategies A, we write $\Delta(A)$ to mean the corresponding set of mixed strategies.

2 The Model

The game at hand is a two-player repeated game with vector payoffs in continuous time. We assume that the players use *nonanticipative behavior strategies* with delay. This means that the behavior of a player may depend only on past play.

Let $A = \{1,2\}$ be a discrete set, $a_i : [0,T] \to A$ a measurable function of time and $a_j : [0,T] \to A$ a random disturbance. Let $u : A \times A \to M$ where M is a 2×2 bimatrix (each entry is a two-dimensional vector). Let $X := conv\{M_{lk} | \, l,k \in A\}$, where *conv* denotes the *convex hull*, and consider the differential equation in X

$$\begin{cases} dx(t) = \frac{1}{t}(\mathbb{E}u(a_i(t),a_j(t)) - x(t))dt, & \forall t \in [0,T], \\ x(0) = x_0 \in X, \end{cases} \tag{1}$$

where $\mathbb{E}u(a_i(t),a_j(t))$ is the expected payoff, and x_0 is generated according to a distribution law $m_0(x)$. More specifically, consider a probability density function $m : X \times [0,+\infty[\to \mathbb{R}$, $(x,t) \mapsto m(x,t)$, representing the density of the players whose state is x at time t, which satisfies $\int_{\mathbb{R}} m(x,t)dx = 1$ for every t. Let us also define the mean state over players at time t as $\overline{m}(t) := \int_X xm(x,t)dx$. We also have $m(x,0) = m_0(x)$.

The objective of a player is to approach a given target $y : [0,T] \to X$. Then, for each group, consider a running cost $g : X \times X \to [0,+\infty[$, $(x,y) \mapsto g(x,y)$ of the form:

$$g(x,y) = \frac{1}{2}\left[(y-x)^T Q\,(y-x)\right], \tag{2}$$

where $Q > 0$ and symmetric.

The above cost describes (i) the (weighted) square deviation of an individual's state from the target.

Also consider a terminal cost $\Psi : X \times X \to [0, +\infty[$, $(x, y) \mapsto \Psi(x, y)$ of the form

$$\Psi(x, y) = \frac{1}{2}(y - x)^T S(y - x), \tag{3}$$

where $S > 0$. The problem in its generic form is then the following:

Problem 1 Let the initial state x_0 be given and with density m_0. Given a finite horizon $T > 0$, a suitable running cost: $g : X \times X \to [0, +\infty[$, $(x, y) \mapsto g(x, y)$, as in (2); a terminal cost $\Psi : X \times X \to [0, +\infty[$, $(y, x) \mapsto \Psi(y, x)$, as in (3), and given a suitable dynamics for x as in (1), solve

$$\inf_{a_i(\cdot) \in \mathscr{C}} \left\{ J(x_0, a_i(\cdot), a_j(\cdot)) = \int_0^T g(x(t), y)dt + \Psi(x(T), y) \right\}, \tag{4}$$

where $\mathscr{C}$ is the set of all measurable functions $a_i(\cdot)$ from $[0, +\infty[$ to A_i, and $\mathbb{E}u(\cdot)$ in (1) must be consistent with the evolution of the distribution $m(\cdot)$ if every player behaves optimally.

3 Main Results

This section outlines the main result of this paper. After introducing the *expected value of the projected game*, Theorem 1 establishes conditions for approachability in 1st-moment.

3.1 Expected Value of the Projected Game

We wish to analyze convergence properties in the space of distributions of the cumulative or average payoff $x_i(t)$, in the spirit of approachability. We will make use of the notion of *projected game* which we recall next. Let $\lambda \in \mathbb{R}^m$ and denote by $\langle \lambda, G \rangle$ the one-shot zero sum game whose set of players and their actions are as in game G, and the payoff that player j pays to player i is $\lambda^T u(a_i(t), a_j(t))$ for every $(a_i(t), a_j(t)) \in A_i \times A_j$. Observe that, as a zero-sum one-shot game, the game $\langle \lambda, G \rangle$ has a *value*, $val(\lambda)$, obtained as

$$val(\lambda) := \min_{a_i(t)} \max_{a_j(t)} \lambda^T u(a_i(t), a_j(t)).$$

Given the stochastic nature of $a_j(t)$ the above min-max operation is not useful to our purposes. Then, we rather consider the expected value of the game (where the inner maximization is replaced by an expectation) and discuss approachability in expectation. In the light of this, and using the bilinear structure of the utility function, and assuming *Markovian strategies*

$$\sigma : X \times [0, T] \to A \text{ such that } a_i(t) := \sigma(x, t),$$

we can rewrite the expected value as

$$\begin{cases} \mathbb{E}val(\lambda) & := \min_{a_i(t)} \mathbb{E}\lambda^T u(a_i(t), a_j(t)) \\ & = \min_{a_i(t)} \lambda^T u(a_i(t), q(t)), \\ \\ q \in \Delta(A) \; s.t. & q_k = \int_{R_k} m(x,t)dx, \\ & R_k := \{x \in \mathbb{R}^m | \, \sigma(x,t) = k\}, \; \forall k \in A, \end{cases} \tag{5}$$

where $\Delta(A)$ is the set of mixed strategies on A. Note that rewriting $\mathbb{E}\lambda^T u(a_i(t), a_j(t))$ as $\lambda^T u(a_i(t), q(t))$ follows from the bilinear structure of the utility function. In the case of state-dependent payoff, which occurs when we consider the game whose payoff is

$$f(u(a_i(t), a_j(t)), x(t)) = \frac{1}{t}(\mathbb{E}u(a_i(t), a_j(t)) - x(t)) = \frac{1}{t}(u(a_i(t), q(t)) - x(t)),$$

the above expression can be modified as:

$$\begin{cases} \mathbb{E}val_x(\lambda) & := \min_{a_i(t)} \mathbb{E}\lambda^T f\Big(u(a_i(t), a_j(t)), x_i)\Big) \\ & = \min_{a_i(t)} \lambda^T f\Big(u(a_i(t), q(t)), x_i\Big) \\ \\ q \in \Delta(A) \; s.t. & q_k = \int_{R_k} m(x,t)dx, \\ & R_k := \{x \in \mathbb{R}^m | \, \sigma(x,t) = k\}, \; \forall k \in A. \end{cases} \tag{6}$$

Note that here we use the notation $u(a_i(t), q(t))$ to mean $\mathbb{E}u(a_i(t), a_j(t))$.

3.2 *Approachability in 1st-Moment*

Approachability theory was developed by Blackwell in 1956 [4] and is captured in the well-known Blackwell's Theorem. We recall next the geometric (approachability) principle that lies behind Blackwell's Theorem.

To introduce the approachability principle, let Φ be a closed and convex set in $\mathbb{R}^m$ and let $P_\Phi(x)$ be the projection of any point $x \in \mathbb{R}^m$ (closest point to x in Φ).

Definition 1 (Approachable Set) A closed and convex set Φ in $\mathbb{R}^m$ is *approachable* by player 1 if there exists a strategy for player 1 such that (7) holds true for every strategy of player 2:

$$\lim_{t\to\infty} dist(x(t), \Phi) = 0. \tag{7}$$

The next result is the Blackwell's Approachability Principle.

Proposition 1 (Blackwell's Approachability Principle [4, 20]) *A closed and convex set Φ in $\mathbb{R}^m$ is approachable by player 1 if for every $x(t)$ there exists a strategy for player 1 such that (8) holds true for every strategy of player 2:*

$$[x(t) - P_\Phi(x(t))]^T[x(t) - P_\Phi(x(t)) + f(u_i(\sigma(x,t), a_j(t)), x_i(t))] \le 0, \quad \forall\, t. \tag{8}$$

Note that in the above statement, condition (8) is equivalent to saying that (i) for every x taking $\lambda = \frac{x-P_\Phi(x)}{\|x-P_\Phi(x)\|} \in \mathbb{R}^m$ the value of the projected game satisfies

$$[x(t) - P_\Phi(x(t))]^T[x(t) - P_\Phi(x(t))] + \|x - P_\Phi(x)\| val_x(\lambda) \le 0, \quad \forall\, t. \tag{9}$$

Now, if we assume that the opponent is committed to play a mixed strategy $q \in \Delta(A)$, condition (8) turns into

$$[x(t) - P_\Phi(x(t))]^T[x(t) - P_\Phi(x(t)) + f(u(\sigma(x,t), q(t)), x(t))] \le 0, \quad \forall\, t, \tag{10}$$

and the corresponding condition (9) can be rewritten as

$$\begin{cases} [x(t) - P_\Phi(x(t))]^T[x(t) - P_\Phi(x(t))] + \|x - P_\Phi(x)\|\mathbb{E}val_x(\lambda) \le 0, \quad \forall\, t, \\ \mathbb{E}val_x(\lambda) := \min_{a_i(t)} \lambda^T f(u_i(a_i(t), q(t)), x_i). \end{cases} \tag{11}$$

Theorem 1 (Approachability in 1st-Moment) *Let $q \in \Delta(A)$ be given. The set of approachable targets is*

$$\mathscr{T}(q) = \{y \mid y = \sum_{l,k\in A} p_l q_k M_{lk}, \forall p \in \Delta(A)\}.$$

Furthermore, approachable strategies are Markovian and bang-bang:

$$\sigma(x) = \begin{cases} a_i = 1 \text{ if } x \in R_1 := \{\xi \mid (\xi - y)^T(u(1,q) - y) \le 0\} \\ a_i = 2 \text{ otherwise.} \end{cases} \tag{12}$$

In the problem at hand, one additional challenge is that q must be *self-confirmed*. This means that the mixed strategy q entering the computation of the expected value of the projected games $\mathbb{E}val_x(\lambda)$ must reflect the current state distribution. This corresponds to solving:

$$\begin{cases} [x(t)-y]^T[x(t)-y] + \|x-y\|\mathbb{E}val_x(\lambda) \leq 0, \quad \forall\, t. \\ \mathbb{E}val_x(\lambda) := \min_{a_i(t)} \lambda^T f\Big(u(a_i(t), q(t)), x\Big) \\ q \in \Delta(A)\; s.t.\; q_k = \int_{R_k} m(x,t)dx, \\ \qquad R_k := \{\xi |\, (\xi - y)^T(u(k,q) - y) \leq 0\}\; \forall k \in A. \end{cases} \tag{13}$$

In an extended journal version of this paper we look for self-confirmed solutions, which we call equilibria.

4 Conclusions and Future Developments

We have extended approachability to population games. In a future work we will adapt the concept of mean-field equilibrium to our evolutionary set-up; we call this *self-confirmed equilibrium*. We will also explore the regret interpretation of our model; whereas 1st-moment approachability of nonpositive regrets no longer implies Nash equilibrium (as in [9]), we will show that nonpositive maximal regret does imply Bayesian equilibrium under incomplete information. Considering the cumulative payoff rather than the average payoff leads to the new notion of *attainability* as in [3]. We will extend our stochastic analysis to attainability in a population setting.

Acknowledgements The work of D. Bauso was supported by the 2012 "Research Fellow" Program of the Dipartimento di Matematica, Università di Trento and by PRIN 20103S5RN3 "Robust decision making in markets and organizations, 2013–2016". This work has developed during the sabbatical period spent by D. Bauso as academic visitor at the Department of Engineering Science, University of Oxford, UK

References

1. Aumann, R.J.: Markets with a continuum of traders. Econometrica **32**(1–2), 39–50 (1964)
2. Bauso, D., Notarstefano, G.: Distributed n-player approachability and consensus in coalitional games. IEEE Trans. Autom. Control (2014), (in print). Available online at doi: 10.1109/TAC.2015.2411873
3. Bauso, D., Lehrer, E., Solan, E., Venel, X.: Attainability in repeated games with vector payoffs. INFORMS Math. Oper. Res. 6054 (2014) Online version http://dx.doi.org/10.1287/moor.2014.0693
4. Blackwell, D.: An analog of the minimax theorem for vector payoffs. Pacific J. Math. **6**(1), 1–8 (1956)
5. Cesa-Bianchi, N., Lugosi, G.: Prediction, Learning and Games. Cambridge University Press, Cambridge (2006)

6. Foster, D., Vohra, R.: Regret in the on-line decision problem. Games Econ. Behav. **29**, 7–35 (1999)
7. Hart, S.: Adaptive heuristics. Econometrica **73**, 1401–1430 (2005)
8. Hart, S., Mas-Colell, A.: A general class of adaptive strategies. J. Econ. Theory **98**, 26–54 (2001)
9. Hart, S., Mas-Colell, A.: Regret-based continuous-time dynamics. Games Econ. Behav. **45**, 375–394 (2003)
10. Huang, M.Y., Caines, P.E., Malhamé, R.P.: Individual and mass behaviour in large population stochastic wireless power control problems: centralized and Nash equilibrium solutions. In: Proceedings of the 42nd IEEE Conference on Decision and Control, pp. 98–103 (2003)
11. Huang, M.Y., Caines, P.E., Malhamé, R.P.: Large population stochastic dynamic games: closed loop kean-vlasov systems and the Nash certainty equivalence principle. Commun. Inf. Syst. **6**, 221–252 (2006)
12. Huang, M.Y., Caines, P.E., Malhamé, R.P.: Large population cost-coupled lqg problems with non-uniform agents: individual-mass behaviour and decentralized ϵ-nash equilibria. IEEE Trans. Autom. Control **9**, 1560–1571 (2007)
13. Jovanovic, B., Rosenthal, R.W.: Anonymous sequential games. J. Math. Econ. **17**, 77–87 (1988)
14. Lasry, J.m., Lions, P.l.: Jeux à champ moyen. i - le cas stationnaire. C.R. Math. **343**(9), 619–625 (2006)
15. Lasry, J.m., Lions, P.l.: Jeux à champ moyen. ii - horizon fini et controle optimal. C. R. Math. **343**(10), 679–684 (2006)
16. Lasry, J.m., Lions, P.l.: Mean field games. Jpn. J. Math. **2**, 229–260 (2007)
17. Lehrer, E.: Allocation processes in cooperative games. Int. J. Game Theory **31**, 341–351 (2002)
18. Lehrer, E.: Approachability in infinite dimensional spaces. Int. J. Game Theory **31**(2), 253–268 (2002)
19. Lehrer, E.: A wide range no-regret theorem. Game Econ. Behav. **42**, 101–115 (2003)
20. Lehrer, E., Sorin, S.: Minmax via differential inclusion. Convex Anal. **14**(2), 271–273 (2007)
21. Soulaimani, A.S., Quincampoix, M., Sorin, S.: Approachability theory, discriminating domain and differential games. SIAM J. Control. Optim. **48**(4), 2461–2479 (2009)