Springer Series in Advanced Manufacturing

Lyes Benyoucef
Jean-Claude Hennet
Manoj Kumar Tiwari *Editors*

Applications of Multi-Criteria and Game Theory Approaches

Manufacturing and Logistics

Springer

Springer Series in Advanced Manufacturing

Series editor

Duc Truong Pham

For further volumes:
http://www.springer.com/series/7113

Lyes Benyoucef · Jean-Claude Hennet
Manoj Kumar Tiwari
Editors

Applications of Multi-Criteria and Game Theory Approaches

Manufacturing and Logistics

Springer

Editors
Lyes Benyoucef
LSIS
Aix-Marseille University
Marseille
France

Jean-Claude Hennet
LSIS
CNRS
Marseille
France

Manoj Kumar Tiwari
Department of Industrial Engineering
and Management
Indian Institute of Technology
Kharagpur, West Bengal
India

ISSN 1860-5168 ISSN 2196-1735 (electronic)
ISBN 978-1-4471-5294-1 ISBN 978-1-4471-5295-8 (eBook)
DOI 10.1007/978-1-4471-5295-8
Springer London Heidelberg New York Dordrecht

Library of Congress Control Number: 2013948364

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Foreword

The activities of the IFAC Technical Committee (TC) 5.2 on "Manufacturing Modeling for Management and Control" (http://tc.ifac-control.org/5/2) are devoted to promote the development of formal descriptive or prescriptive models of manufacturing systems. IFAC TC 5.2 is composed of 66 members, leading scientists in this domain from 23 countries. There are three working groups: "Supply network engineering", "Advanced multi-criteria applications in manufacturing and logistics", and "Design and modeling of flexible and reconfigurable manufacturing systems". The committee organizes recurrently its conference MIM and co-organizes several other conferences, workshops, special sessions, special issues of refereed journals, etc. The activities of the committee are rich and very fruitful.

The book "Applications of Multi-Criteria and Game Theory Approaches" is a result of a long work of TC and scientific discussions conducted by Prof. Lyes Benyoucef, Dr. Jean-Claude Hennet and Prof. Manoj Kumar Tiwari, Co-Chairs of the TC working groups. It deals with the major current issues of manufacturing management and control: How to handle the increasing complexity of logistic networks? How to design logistic systems taking into account a great number of possible configurations? How to search for trade-off between multiple conflicting criteria? How to coordinate different activities in a distributed and heterogeneous environment? Extremely interesting approaches based on multi-criteria optimization and game theory are suggested in this outstanding book.

Prof. Alexandre Dolgui
Chair of the IFAC Technical Committee 5.2

Preface

The new competition is a major upheaval affecting every aspect of how enterprises organize and operate. The evolution from single enterprise with a high vertical range of activities toward enterprise networks offers new business opportunities especially for small and medium enterprises (SMEs) that are usually more flexible than larger companies. However, in order to make a successful commitment to an enterprise network, expected performance and benefits have to be carefully evaluated and balanced for a company to become a partner of the right network and for the right tasks. All these issues have to be taken into account in order to find an efficient, flexible, and sustainable solution.

In the area of manufacturing and logistics, supply chain networks involve transformation processes from raw materials to finished products, through several stages of manufacturing, assembly, distribution, and delivery to customers. They also rely on information and monetary flows in addition to material flows. Each stage of material transformation or distribution may involve inputs coming from several suppliers and outputs going to several intermediate customers. Furthermore, each stage may involve information and material flows connected with some intermediate and distant stages.

The underlying logistic networks are complex and their analysis requires a carefully defined approach. As technological complexity has increased, logistic networks have become more dynamic and complex to handle. Consequently, there is a strong risk for practitioners and managers to get lost in details and spend a large amount of effort for analyzing the logistic network without meaningful results. Another issue coming along with the design and management of logistic networks is the great variety of possible policies and alternatives for each of these problems (design, management, and operations), and the need to assess complex trade-offs between conflicting criteria such as cost, quality, delivery, flexibility, robustness, etc. Moreover, world class supply chains typically involve different enterprises sharing common information and logistic networks. Due to the distributed nature of the network and the decisional autonomy of heterogeneous decision centers, organization of tasks and activities raises specific problems of coordination and integration. Enterprises can be seen as players in a game defined by a common goal, but separate constraints and conflicting objectives.

Multi-criteria approaches have been put to use in many segments of manufacturing and logistics. They have taken a prominent role to integrate people, information, and products in manufacturing, warehousing, and distribution of goods and services. Decisions involving customer profiling, new product development, retail marketing, and sales patterns can be greatly refined using innovative multi-criteria approaches. Also, as such decisions have an impact on the overall integrated logistic network process, it is important to link the innovative multi-criteria-based tools to most integrated supply chain management applications.

Game theory provides a mathematical background for distributed systems and generating solutions in cooperative, competitive, or conflicting situations. Much effort has been recently devoted to constructing game theory models of supply chains and using them for better design, organization, and performance. One of the contributions of the book will be to describe those achievements in game theory that are particularly relevant and useful in the modern manufacturing world.

This book aims to align latest practice, innovation, and case studies with academic frameworks and theories covering the broad area of multi-criteria and game theory applications in manufacturing and logistics. Sixteen chapters were selected after a peer review process. They were revised in accordance with the suggestions and recommendations from the reviewers. The book comprises two main parts. Part I contains ten chapters (Chaps. 1–10) dedicated to "Multi-criteria Applications". Part II is dedicated to "Game Theory Applications". It contains six chapters (Chaps. 11–16).

Chapter 1, by N. Labadie and C. Prodhon, provides a survey on multi-criteria analysis in logistics with a focus on vehicle routing problems. The chapter highlights most recent key references dedicated to multi-criteria studies in transportation logistics and especially on vehicle routing problems. Moreover, it presents some interesting research directions for future works.

Chapter 2, by X. Delorme et al., deals with the use of multi-objective approaches in the field of assembly line design. The design of assembly lines is a very important industrial problem that involves various difficult and interconnected optimization problems. A review of the main multi-objective optimization methods used for these problems is presented and discussed. A case study is also described in order to highlight some interesting properties associated with such multi-objective problems.

Chapter 3, by E. Çevikcan et al., proposes a multi-objective decision-making approach to select the best storage policy with respect to the company's requirements. After providing background information about storage policies as well as storage assignment models, a fuzzy information approach is proposed for storage policy selection. The approach is validated with the help of an illustrative case study from the automotive industry.

Chapter 4, by S. Mungle et al., develops a thermodynamically inspired high performance multi-objective evolutionary algorithm (TDHP-MOEA) incorporated with quality function deployment (QFD) and fuzzy-analytic network process (FANP) to resolve the product technical requirements (PTRs) selection problem in product design. The proposed approach considers goals such as new product development (NPD) time and cost, technological advancement, and manufacturability for selection of the most suitable PTRs. A case study of software development is presented to demonstrate the effectiveness of the proposed approach.

Chapter 5, by F. Belmecheri-Yalaoui et al., presents two multi-objective meta-heuristic methods based on ant colony algorithm to simultaneously optimize the storage problem and the quayside transport problem observed during resolution of the container terminal problem. The container terminal has to manage container traffic at the crossroads of land, road, and railway. The results of both meta-heuristics are compared with those obtained by a complete enumeration of the solutions.

Chapter 6, by Y. L. Chong et al., proposes an exploratory study to predict the most important factors that can lead to successful mobile supply chain management adoption for manufacturing firms. The results show that some of the strongest predictors for mobile supply chain management adoption are senior management support, security perceptions, technology integration, financial, and technical competence. Moreover, the study shows that firm size and environmental factors have less predictive power than technological and organizational factors on mobile supply chain management adoption decisions.

Chapter 7, by L. Berrah and L. Foulloy, deals with the problem of computing performance expressions in modern industrial companies. Performance expressions are the results of performance indicators and performance measurement systems—PMSs. By revisiting previous works in this field, the authors aim to define a unified framework for such a computation, by integrating the industrial context data. Three parameters are considered, respectively, the declared objective, the acquired measurement, and the performance expression that results from the comparison of the measurement to the objective.

Chapter 8, by I.U. Sari et al., proposes a multi-criteria decision-making approach called fuzzy decision-making trial and evaluation laboratory (DEMATEL) to prioritize the supply chain performance measures. The authors first attempt to prioritize the key performance criteria of the performance measurement system using fuzzy DEMATEL and then investigate the effect of fuzzy linguistic scale in the prioritization of the criteria. Two different scales are tested. The results obtained with the use of different scales are similar to each other showing that fuzzy DEMATEL is robust to minor changes in linguistic variable scale.

Chapter 9, by R. A. Kumar et al., focuses on selecting the route in international intermodal freight transportation network. The problem is complex and comprises the following characteristics: (1) multi-objective: minimization of travel time and travel cost, (2) schedules and delivery times of every service provider in each pair of location, and (3) variable cost must be included in every location. First, they formulate the problem into mixed integer linear programming (MILP) model. Second, they develop two different methods, respectively, 'NP' (nested partition) method and 'MADM' (multi attribute-decision making) method to obtain the optimal route. To show the performances of both methods, they present several numerical experiments and discuss the results.

Chapter 10, by N. Labadie et al., addresses the bi-objective multiple traveling salesman problem with profits (BOMTSPP), generalizing the classical TSP with profits (TSPP). This new problem generalizes the TSPP in two directions: a true bi-objective treatment and the construction of multiple tours. Two criteria are considered: the length of the tour, like in the classical TSP, and profits which can be collected at customers. An evolutionary algorithm based on NSGA-II, reinforced by a post-optimization procedure based on path-relinking (PR), is developed. To demonstrate its efficiency, rich experimental results are presented and discussed.

Chapter 11, by D. Xu et al., presents a hybrid simulation-based framework to address duopoly game under the scenario of product adoption process considering multiple decision variables and detailed payoffs. In the proposed hybrid simulation framework, system dynamics (SD) models are used for simulating the activities of duopoly companies on production, logistics, and price determination, where agent-based simulation (ABS) is used for modeling consumer purchasing behaviors at the market side. To illustrate the applicability of the proposed framework, a hypothetical case scenario involving soft-drink duopoly on Coke and Pepsi is presented and numerical results discussed.

Chapter 12, by M. Mateo and E. H. Aghezzaf, discusses the problem of integrating inventory and distribution optimization together with game theory to effectively manage supply networks. The problem is known as the inventory routing problem (IRP) and is an underlying optimization model for supply networks implementing a vendor managed inventory (VMI) strategy. The authors concentrate on the stream of literature focusing on cooperative game theory in the inventory routing problem. Morover, they present and discuss two applications issued from cooperation in the wine distribution and cost allocation in the gas distribution.

Chapter 13, by D. K. Verma et al., considers the problem of determining an optimal set of winning suppliers in a single buyer procurement auction scenario. The buyer wishes to procure high volumes of a homogeneous item in a staggered way in accordance with a predefined schedule and the suppliers respond with bids that specify volume discounts and also delivery lead times. The authors show that the winner determination problem, which turns out to be a multi-objective optimization problem, cannot be satisfactorily solved by traditional methods of multi-objective optimization. They formulate the problem first as an integer program with constraints capturing lead time requirements and show that the integer program is an extended version of the multiple knapsack problem. Moreover, they discover certain properties of this integer program and exploit the properties to simplify it to a 0–1 mixed integer program (MIP), which can be solved more efficiently. Next, they explore a more efficient approach to solving the problem using a linear relaxation of the 0–1 MIP in conjunction with a greedy heuristic. Using extensive numerical experimentation, the efficacy of the 0–1 MIP and the proposed heuristic are demonstrated.

Chapter 14, by S. Mahjoub and J. C. Hennet, analyzes the process of forming a coalition within a corporate network. The objective of the partner companies is to create a multi-stage manufacturing system, which generates a chain of increased value from raw materials to end-user market. This process is studied by cooperative game theory, through the key problems of maximizing the total profit and distributing it among the members of the coalition. To construct a pay-off policy that is both stable and fair, the chapter proposes to represent the productive resources of the companies not only by their capacity, but also by the work in progress (WIP) generated by product flows. The proposed profit sharing rule is then constructed from the dual of the profit maximization problem. It is both efficient and rational, with more fairness than the Owen set policy of classical linear production games.

Chapter 15, by A. El Omri et al., deals with the coalition formation problem in supply chains. The coalition formation problem has a profit maximizing objective, that is, increasing the benefit or the savings a supply chain agent (player/partner) can make by coordinating his activities with other agents. Cooperative game theory setting is used to analyze supply chain situations where a set of independent and freely interacting agents can benefit by working jointly. The authors consider the hedonic settings to study the formation of stable coalition structures in inventory games with general cost function. The goal is to focus on the problems of (i) coalition structure generation, i.e., formation of coalition structures, such that agents inside a coalition coordinate their activities, but agents of different coalitions will work independently; and (ii) worth sharing, i.e., distribution of the worth generated by the coalition to its agents.

Chapter 16, by T.S. Chandrashekar and Y. Narahari, models the multiple units, single item procurement network formation problem as a surplus maximizing network flow cooperative game. The buyer has a demand for a certain number of units. The agents in the network must coordinate themselves to meet this demand. Each edge is owned by a rational utility maximizing agent, where each agent has a capacity constraint on the number of units that he can process. The authors first investigate the conditions under which the core of this game is non-empty, and then, construct an extensive form game to implement the core whenever it is non-empty.

We hope that you will enjoy the results of these efforts.

Lyes Benyoucef
Jean-Claude Hennet
Manoj Kumar Tiwari

Acknowledgments

We wish to place on record our special thanks to Series Editor Prof. D. T. Pham and Senior Editorial Assistant Miss Grace Quinn for their valuable guidance and support during the entire process of editing the book. We offer our thanks to the Springer editorial team for their active role and support.

We would like to thank all reviewers for providing in-depth comments and constructive criticisms, and the authors for contributing their high-quality manuscripts. Without your help it would have been impossible to produce this book.

We hope you will enjoy the result of these efforts.

Lyes Benyoucef
Jean-Claude Hennet
Manoj Kumar Tiwari

Contents

Part I
Multi-criteria Applications

Chapter 1
A Survey on Multi-criteria Analysis in Logistics: Focus on Vehicle Routing Problems

N. Labadie and C. Prodhon

Abstract Vehicle routing problems play a central role in logistics. These combinatorial optimization problems have attracted more and more attention these last five decades both in theory and in practice. However, main contributions are dedicated to the single criterion optimization problems. The goal of this chapter is to provide the recent key references dedicated to multi-criteria studies in transportation logistics and especially on vehicle routing problems and to present some interesting research directions.

Keywords Transportation · Logistics · Vehicle routing · Multi-criteria problems

1.1 Introduction

Vehicle routing problems (VRPs) are combinatorial optimization problems that appear in relevant practical applications covering many different domains from the distribution of goods to the delivery of services. The goal is to build one or several vehicle routes in order to service a set of customers. This family of combinatorial optimization problems has attracted widespread research in the past decades. Indeed, it arrives at the top list of the more studied fields in operations research (Laporte 2009).

Routing problems are usually solved through a single objective aiming to minimize a cost, whereas improvements on the solution cost often have a direct impact on other important factors. Indeed, in real-world applications, one can be interested by optimizing simultaneously other criteria such as fleet size, work balancing, or customer satisfaction.

N. Labadie (✉) · C. Prodhon
ICD/LOSI, Université de Technologie de Troyes, 12 rue Marie Curie,
CS 42060 10004 Troyes Cedex, France
e-mail: nacima.labadie@utt.fr

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*,
Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_1,

The aim of this survey is to provide an overview on routing problems for which more than one objective function must be optimized. Boffey (1995) made a first classification of these topics and presented several useful solution methods. Later, Jozefowiez et al. (2008b) proposed an updated review based on objectives, then on problems, and finally on methods. Our contribution is to refer mainly to papers dealing with vehicle routing which have been published over the last half decade (i.e., those appeared since the last survey from Jozefowiez et al. (2008b)) and to classify them according to the kind of problem. A short review on other multi-criteria problems encountered in the logistic field and involving routing decisions such as shortest path computation or distribution network design is also presented.

This chapter is organized as follows: the classical vehicle routing problem and an overview on multi-objective combinatorial optimization are introduced in the Sect. 1.2. Section 1.3 provides an overview on multi-criteria studies involving the basic vehicle routing problem as a central part. In Sect. 1.4, the principal studies dealing with complex constraints and/or unusual criteria are given. Section 1.5 is dedicated to recent trends on routing problems involving path, flow, or network design in a multi-criteria environment. Section 1.6 provides a classification of the published literature on the subject. A conclusion and some future directions for research are drawn in Sect. 1.7.

1.2 Background

Before presenting the papers related to the review proposed here, it might be necessary to give some settings of the problems under consideration. Thus, this section first recalls the capacitated vehicle routing problem (CVRP) and introduces some notions on multi-objective optimization.

1.2.1 Basic Vehicle Routing Problem

Capacitated Vehicle Routing Problem involves the routing of vehicles with common limited capacity from a central depot to a set of customers at minimal cost. It can be modeled by a complete graph $G = (X, A)$ where $X = \{0,1,\ldots,n\}$ is a set of vertices, and $A = \{(i, j) \mid \forall i, j \in X,\ i \neq j\}$ is a set of arcs. Vertex 0 corresponds to the depot where is based a homogeneous fleet of vehicles with a limited capacity W. The remaining n vertices are the customers. Each customer i has a known demand q_i. Each arc (i, j) is associated to a value $d_{i,j}$ which represents the cost of the shortest path linking the nodes i and j. This value can be a monetary cost, a distance, a time, etc. The aim is to build a set of routes with a minimal total cost servicing each customer exactly once, without exceeding the vehicle capacity. The CVRP has been proved NP-hard (Lenstra and Rinnooy Kan 1981), for an overview on mathematical formulations, exact and approximate methods designed to solve this problem; see for example, Toth and Vigo (2002) and Laporte (2009).

Some other surveys dealing with more complex variants of the vehicle routing problems have appeared recently in the literature. Hoff et al. (2010) provide an overview dedicated to routing and fleet composition problems, where the fleet is composed of several types of vehicles associated with different fixed and variable costs. This last paper focuses on aspects related to industrial applications. Labadie and Prins (2012) present also a survey summarizing the most important results on the majority of vehicle routing variants, with an emphasis on problems occurring in developing countries. In Baldacci et al. (2012), mathematical formulations, relaxations, and recent exact methods developed to resolve the CVRP and the VRP with time windows (VRPTW) are given. VRPTW is the most widely studied variant of the CVRP and differs from this last on the fact that for each customer is associated a time slot within which its service must start. Classification schemes as well as exact and heuristic algorithms are given in Nagy and Salhi (2007) for the location-routing problem. In this relatively recent category of problems, simultaneously to routing decisions one looks on how to locate optimally the depots from which the customers would be serviced. Capacitated arc routing problem (CARP) is the arc counterpart of the CVRP in the sense that focus regarding service and resource constraints are on the links and not on the nodes of the given graph. This routing problem is much less studied in the literature inspite of its numerous applications such as electrical lines inspection, snow removal, garbage collection, etc. For an extended survey on this problem, its variants, formulations, and resolution approaches see the paper by Wøhlk (2008).

Contrarily to the problems cited above, in routing problems with profits it is not necessary to service all the customers. In this branch of problems, each customer (node) is associated to a positive score or profit which is collected only if the corresponding node is visited. Interested readers are referred to Feillet et al. (2005) and to Vansteenwegen et al. (2011) for a survey on the different categories of problems and the corresponding results appeared in the literature.

In covering tour problems, some locations must just be covered and not necessarily visited. Such kind of problems have many applications in delivery services such as health care to rural population in developing countries. The aim is to build a tour visiting some centers with a minimal total length in order to guarantee coverage of a set of customers (population). This notion of coverage is often associated to a given distance, which is considered as a problem parameter. These problems are by nature multi-criteria since at least they can aim to minimize the tour length or cost, maximize the population covered, and minimize the maximal distance to a center included in the tour.

Most published papers on problems involving routing problems concern the single objective case. Multi-objective studies attract very less attention, although in real-word applications several objective functions are often expressed. This chapter targets to gather recent studies published since the last survey by Jozefowiez et al. (2008b) on the most important vehicle routing problems generalizing the basic CVRP and CARP, and especially the problems cited above. Recent developments on some other problems involving routing decisions are also mentioned in this survey.

1.2.2 Multi-objective Optimization

In single objective optimization, the goal is to find one solution (or in special cases multiple optimal solutions but with the same objective function value). In multi-objective optimization, this is not sufficient since problems deal with more than one objective function constituting a multi-dimensional objective space. The aim is then to find the set of so-called Pareto-optimal solutions or efficient solutions. A feasible solution x_1 is called efficient if there does not exist another feasible solution x_2 such that the value of x_2 is better or equal to the value of x_1 for all objective functions, with a strict inequality for at least one of the objectives. Otherwise, x_2 dominates x_1.

The main goal in multi-objective optimization is to find a set of solutions that approximates well the Pareto-optimal set (or the non-dominated vectors in the objective space), i.e., (1) as close as possible to the Pareto-optimal front and (2) as diverse as possible to guarantee a good set of trade-off solutions.

A first approach is to transform and solve a single objective problem through a weighted metric method that scalarizes the set of objectives. The resulting solutions are defined as the set of supported efficient ones, *SE*. However, a routing problem is usually combinatorial leading to a multi-objective combinatorial optimization (MOCO). The fact to deal with discrete variables has a strong consequence on the difficulty of such problems. Although the objectives are usually linear functions, there may exist efficient solutions, called non-supported efficient solutions *NE* which are not optimal for any weighted sum of the objectives. Finding the non-supported solutions contributes essentially to the difficulty of MOCO problems. Thus, a two-phase method can be applied. In the first phase, *SE* is found using the scalarization technique, and solving single objective problems. The second phase consists of finding the non-supported efficient solutions by problem-specific methods using bounds, reduced costs, etc.

Another approach is the adaptation of metaheuristic techniques. A first kind consists of defining search directions by a local aggregation of the objectives, often based on a weighted sum. Thus, starting from an initial solution and a given direction, an approximation of a part of the Pareto-optimal front can be found. The principle is repeated on several directions to retrieve completely the non-dominated frontier. A second kind is based on both a population of solutions and the notion of dominance to approximate the non-dominated frontier. It has the advantage of searching for many efficient solutions per iteration. Finally, there exist also specific procedures and hybrid methods.

For more details and guidelines on the development and use of the most effective metaheuristics methodologies for MOCO see, for example, Deb et al.

(2000), Deb (2002), Angus and Woodward (2009). The two first papers are dedicated to multi-criteria evolutionary algorithms and the last one deals with ant colony systems. In Martí et al. (2011), a methodology for adapting the hybrid metaheuristic greedy randomized adaptive search procedure (GRASP) combined with the path relinking approach (PR) is developed for multi-criteria problems. The most frequently used resolution approaches are based on multi-criteria evolutionary algorithms which are detailed through numerous surveys such as the papers by Coello Coello et al. (2005), Coello Coello (2009), Zhou et al. (2011).

1.3 Multi-criteria Analysis for the Basic Routing Problem

This section aims to make a census of multi-criteria studies involving the basic routing problem, i.e., problems without extra constraints or attribute. The studies cited here consider the CVRP defined above as a core problem but add one or more criteria which must be optimized simultaneously in addition to a cost function.

The study by Parc and Koelling (1986) is the pioneer one dealing with multi-objective CVRP. In this work, the classical CVRP is considered with three conflicting criteria: minimization of the total distance traveled; minimization of the total deterioration of goods during transportation; and maximization of fulfillment of emergent services and conditional dependencies of customers. This third criterion is relevant for cases where some customers should be serviced urgently or are contingent upon others. Two customers are said to be contingent when there is a conditional dependency between them; these dependencies could be resulting from operational, functional, or economic reasons. The problem is resolved using heuristics that take into account the decision makers' preferences.

Since then, many papers have been devoted to this issue. From the recent years, we can quote Jozefowiez et al. (2009) who consider a CVRP in which the total route length and the route imbalance are minimized concurrently. The second criterion in this study consists in minimizing the difference between the longest route and the shortest one. A multi-objective evolutionary algorithm using a new mechanism, called the elitist diversification, is used in cooperation with a sharing method and parallelization techniques to resolve the problem. In a previous study, these authors (Jozefowiez et al. 2005) considered the same problem and resolved it with an enhancement of the popular NSGA-II (Non-dominated Sorting Genetic Algorithm).

The study by Chand et al. (2010) deals with a bi-criteria CVRP in which the number of vehicles and the total cost (distance) are minimized. A genetic algorithm-based approach is designed to resolve this problem; however, in this study the authors do not look for a Pareto front but for a single solution. This one has to be optimized for each objective so that, if we try to optimize it any further, the other objective(s) will suffer as a result. The approach is tested using problem instances reported in the literature, derived from publicly available Solomon's benchmark data for VRP. According to the authors, the results show that the GA approach is able to find high quality solutions but unfortunately they do not provide comparisons with previous studies.

For the arc counterpart of the CVRP, namely the capacitated arc routing problem (CARP), we are only aware of one study due to Lacomme et al. (2006). In this work, in addition to the frequently used criterion which is the total cost of the trips, a second criterion related to the makespan, as in scheduling problems, is minimized conjointly. This second objective function consists in minimizing the longest trip and the bi-criteria CARP is solved thanks to an efficient non-dominated sorting genetic algorithm (NSGA-II).

1.4 Multi-criteria Analysis for VRP with Rich Structure

1.4.1 Vehicle Routing with Time Windows

Vehicle Routing Problem with Time Windows extends the basic CVRP by adding time constraints on customers' service. In this variant, to each customer i is associated a predefined time lag $[b_i, e_i]$, within the service must start. A time window $[b_0, e_0]$ is often considered for the depot's opening hours, and traveling times t_{ij} are defined in addition to distances $d_{i,j}$. Due to its academic interest and its numerous real-life applications (such as in maintenance routing problems), VRPTW is drawing more and more attention in the research community. Most of the published literature deals with hard time windows. In this case, when a vehicle arrives at customer i before b_i, it has to wait and it is not allowed to service a customer after the closing time e_i. In some versions, late and/or early services are permitted but penalty costs must be paid (soft time windows). Contrary to the CVRP, deciding whether m routes are enough to visit all customers is an NP-complete problem. Most authors minimize the number of vehicles required and then the total distance performed; traveling times are just used to check time windows. The VRPTW is NP-Hard and instances with 100 customers or more are very hard to solve optimally. The majority of resolution methods are approximations, and evolutionary algorithms account for the greater part.

When the number of vehicles is to be minimized in priority, the best metaheuristics are the memetic algorithm of Nagata et al. (2010) and the arc-guided evolutionary algorithm of Repoussis et al. (2009). Labadie et al. (2008) design an effective memetic algorithm for total distance minimization, as in the CVRP. The same algorithm is also able to resolve efficiently the problem where the number of vehicles must be minimized in priority before the total distance. For a complete overview on resolution approaches for the VRPTW, one can see the surveys of Bräysy and Gendreau (2005a, b).

The multi-criteria version of the VRPTW is without any doubt, the most investigated among multi-objective vehicle routing problems. Rahoual et al. (2001) design an NSGA-based genetic algorithm for the VRPTW for minimizing the number of routes, the travel distance, and the penalties associated with violated constraints.

Tan et al. (2006) and Ombuki et al. (2006) consider the VRPTW as a bi-objective optimization problem, minimizing the number of vehicles and the total travel distance. Both studies propose a genetic algorithm for solving the problem and use the standard Solomon's benchmark to assess the quality of the developed approaches (see Solomon (1987) for more details). In the former study, a Pareto ranking techniques is used to assign fitness to individuals, design a new crossover operator called route-exchange crossover, and use a multi-mode mutation which considered swapping, splitting, and merging of routes. The latter propose the genetic operators best cost route crossover and constrained route reversal mutation, which is an adaptation of the widely used inversion method.

In Ghoseiri and Ghannadpour (2010), the same problem as in Tan et al. (2006) and Ombuki et al. (2006), is studied. The authors propose a goal programming approach and a genetic algorithm in which the decision maker specifies optimistic aspiration levels to the objectives and deviations from those aspirations are minimized. The method is applied to solve Solomon's benchmark of 56 VRPTW instances with 100 customers. The results are compared to the best known solutions obtained for the single objective case or to the two previous studies cited above and are proved to be competitive.

In the study of Wang and Li (2011), a multi-objective VRP considering time window constraints is also investigated. The authors consider two objective functions, the first consists in minimizing the total distance while the second maximizes client satisfaction by fulfilling time-window requirements. A hybrid genetic algorithm was designed to solve the problem; the numerical evaluations of this method are driven on a military application.

Garcia-Najera and Bullinaria (2011) study the VRPTW with three criteria to minimize: the total crossed distance, the overall traveling time, and the fleet size. This paper proposes a multi-objective evolutionary algorithm, which incorporates methods for measuring the similarity of solutions, to solve the problem. The numerical results obtained on Solomon's instances show that when the similarity measure is used, the diversity and the quality of solutions are improved. Furthermore, the algorithm achieves competitive results since it provides better Pareto front approximations.

In Muller (2010), a VRP with soft time windows (VRPSTW) is considered. That means violations of the time windows are allowed, but associated with penalties. The problem studied resides in determining optimally the routes so as to minimize simultaneously the total costs, consisting of the number of used vehicles and the total distance, on one part and the penalties on the other part. The problem is formulated as a bi-criteria minimization problem and heuristic methods are used to calculate approximations of the Pareto optimal solutions. Experimental results show that in certain cases the allowance of penalties leads to significant savings of the total costs.

Tavakkoli-Moghaddam et al. (2005) consider the VRPSTW with a heterogeneous fleet of vehicles. Three criteria are to minimize: fleet cost, routes cost, and violation of soft time windows penalty. The authors use a simulated annealing (SA) approach with the classical 1-Opt and 2-Opt operators for solving the problem. More recently, Tavakkoli-Moghaddam et al. (2011) have considered again a new variant of the VRPTW with two objective functions to optimize. The authors call this problem a VRP with competitive time windows (VRPCTW) and the considered criteria are the total traveling time to minimize and the total amount of sales to maximize. In this new problem that occurs in a competitive environment, the demand of each customer is constituted of two parts, the first part does not depend on time and should be delivered completely to the customer, the second part is time-dependent and would be lost if the rival's arrival time is earlier than vehicle's arrival time to the customer. A new mathematical model is developed for the proposed problem and for solving it and a simulated annealing approach is used. The small test problems are solved by the SA and the results are compared with obtained results from Lingo software. For large-scale problems, Solomon's benchmark instances with additional assumption were used and SA algorithm was able to find good solutions in reasonable time.

Norouzi et al. (2009) present also a study dealing with a routing problem under competition. In this study, there is no time-window, but still time-dependent constraints. More precisely, the authors propose a mathematical model for a bi-objective open vehicle routing problem in a competitive environment (OVRPC). This problem consists of a VRP for which the routes do not return to depot after the last customer. In addition, it is supposed that the profit made at a customer depends on the time on which it is visited, i.e., if a vehicle visits a customer later than its rival, it will miss a part of its sale. Hence, in order to maximize the profit, the company should serve customers earlier than its rival while minimizing the total length of the routes. The authors propose a multi-objective particle swarm optimization (MOPSO) method, a population-based approach inspired from the behavior of natural group organisms, such as bees, fishes, and birds swarm. The results are compared with the Lingo software using a ε-constraint method on small-sized test problems.

Gupta et al. (2010) study a multi-objective fuzzy vehicle routing problem with time windows and capacity constraints (MOFVRP). The concept of fuzzy logic is used to deal with uncertainty on traveling time between two stops and a genetic algorithm is used to deal with multiple attributes: maximization of customer's satisfaction grade, minimization of fleet size, distance minimization, and waiting time minimization. To demonstrate the effectiveness of the developed approach, a case study is used. It concerns a bus collection application where students must be picked-up and dropped from/to university in India.

Braekers et al. (2011) consider a full truckload vehicle routing problem with time windows encountered in drayage operations. Loaded and empty container transports are to be performed where either the origin or the destination of empty containers must be determined. The authors show that this problem can be transformed into an asymmetric multiple vehicle traveling salesman problem with time windows (am-TSPTW) and a two-phase deterministic annealing algorithm is developed for solving the problem in which the number of vehicles used is minimized as well as the distance traveled. The first phase of the method consists in minimizing the fleet size and the second, the total distance for the current number of vehicles. Deterministic simulated annealing metaheuristics are used in both phases and the performance of global method is tested on randomly generated instances.

1.4.2 Vehicle Routing with Several Depots

In multi-depot (MD) problems, the departure and return nodes for each vehicle must be selected among a set of depots. The first case considers uncapacitated depots and leads to the MDVRP. Lau et al. (2009) study a multi-objective version with multiple products for which the aim is to minimize both the total traveling distance and the total traveling time required for all vehicles. They propose a fuzzy logic guided NSGA-II (FL-NSGA-II). The role of fuzzy logic is to dynamically adjust the crossover and mutation rates after consecutive generations. They compare their method with a classical NSGA-II, but also with a strength Pareto evolutionary algorithm 2 (SPEA2) and a micro-genetic algorithm (MICROGA), each time with and without the guide of fuzzy logic. The results show that FL-NSGA-II outperformed other search methods on the tested scenarios.

The second case of MD problems occurs when depots are capacitated and/or when the location of those is a decision variable. Location of facilities and vehicle routing, when studied and solved commonly, constituted the location-routing problem (LRP). Nagy and Salhi (2007) have made a survey on the subject. Since then, some papers have been published on the mono-objective case. Prins et al. (2007) and Duhamel et al. (2010) propose the current best efficient metaheuristics and recently, Belenguer et al. (2011) introduced mathematical models and exact solutions methods but they are still limited to medium-scale instances. Prodhon (2011) also studies a periodic version. However, in the past, multi-objective versions were often discarded. Only Lin and Kwok (2006) addressed the case in which total cost minimization and workload balance were the objectives. Additionally, in this study a version with multi-route consideration was possible during the routing procedure. The authors applied two metaheuristics (tabu search and simulated annealing) on real and simulated data and compared the results of two versions: simultaneous or sequential routes assignment to vehicles. Other papers were published for hazardous transportation, in which apart from the cost, the location and/or a transportation risk have to be minimized to ensure a safety perimeter for the population (List and Mirchandani 1991; Giannikos 1998; Alumur and Kara 2007).

Nowadays, criteria to optimize in addition to the total cost are more related with the demand to be served. Tavakkoli-Moghaddam et al. (2010) present a new integrated mathematical model for a bi-objective version where the total cost (setup cost of the facility, fixed and variable depot costs, and routing cost) has to be minimized while the total demand to be served has to be maximized.

The authors propose a multi-objective scatter search (MOSS) algorithm and validate both the solution quality and diversity level on various test problems through some comparison metrics with the elite tabu search (ETS).

In the same vein, an interesting application of multi-objective LRP concerns logistics of relief. Rath and Gutjahr (2011) consider a problem faced after the occurrence of a natural disaster. A supply system with intermediate warehouses has to be established to provide affected people with relief goods. It may happen that total supply is less than total demand. Thus, a three-objective optimization model is proposed. The first objective minimizes the fixed costs for depots and vehicles. The second objective minimizes operative cost (routing and warehousing). The third objective maximizes the covered demand. They apply the ε-constraint method to determine the Pareto frontier and solve the single-objective problem by a metaheuristic technique based on an MILP formulation with a VNS algorithm to iteratively add heuristically generated constraints. Results on generated instances and a real case are compared to those obtained from an application of the NSGA-II metaheuristic.

Coutinho-Rodrigues et al. (2012) also study multi-objective catastrophe responses for urban evacuation paths and location of shelters. Six objectives are considered in an MILP model, including the minimization of total travel distance for all of the population to shelters, the minimization of the risk on paths and at the shelters, and the minimization of the total time required to transfer people from shelter to a hospital. The proposed approach is tested for a simulated fire situation in the historical city center of Coimbra, Portugal. The solutions are compared in the objective space via several graphical techniques.

1.4.3 Routing Problems with Profit

In routing problems with profits, for each customer a positive profit (score) is given, in addition to the elementary data defining a basic CVRP (graph *G*). In some variants, a penalty can also be associated to each customer. These kinds of problems permit to visit only a subset of customers and occur in industrial application such as scheduling repairmen visits to the most profitable customers, tourist travel guide systems, etc. This family of routing problems with profits is by nature multi-objective with two opposite optimization criteria. The first objective consists in maximizing the total profit; it hence forces to extend the tour and collect as much profit as possible increasing therefore the traveled distance. The second criterion, in opposition with the first one, instigates to reduce the total traveled distance and consequently tends to visit fewer customers. In spite of the bi-objective nature of this category of problems, the research has been mostly focused on the mono-criterion case.

The variant where only one tour has to be determined is referred to as the traveling salesman problems with profits (TSPP). Feillet et al. (2005) discuss three generic problems derived from the TSPP, depending on how the two objectives are tackled. In the first one, both criteria are expressed in the objective function which consists in minimizing the travel costs minus the collected profit. This problem is called profitable tour problem (PTP). In the second class, the travel costs are expressed as a constraint. The profit is maximized while the length of the tour must not exceed a given limit. This problem is called the orienteering problem (OP). In the third class, the total profit is expressed as a constraint and it must not be less than a given value and the travel costs are minimized. This last variant is referred to as prize collecting traveling salesman problem and often considers penalties on the customers not serviced. The sum of these penalties (when defined) is then added to the total traveled distance to obtain the objective function, which must be minimized.

The team orienteering problem (TOP) is an extension of the orienteering variant to the case where a fixed number (great or equal to 2) of tours must be built. TOP has been defined for the first time by Chao et al. (1996) and is, besides the orienteering variant (OP), the more studied problems among all those cited above. However, most published papers in the literature focus on the mono-objective variants.

The multi-objective version of the TSPP has been considered for the first time in Keller and Goodchild (1988). After this first study, to the best of our knowledge, only four journal papers have been published. The first is from Riera-Ledesma and Salazar-González (2005) who study the traveling purchaser problem, in which the nodes represent markets of different products. The traveling purchaser must visit a subset of markets in order to purchase the required quantity of each product while the travel cost and the purchase cost are both minimized. In Jozefowiez et al. (2008a), an ejection chain local search enhanced within a multi-objective evolutionary algorithm is developed to generate efficient solutions to the traveling salesman problem with profits. Bérubé et al. (2009) propose an exact ε-constraint method for the same problem and finally, Schilde et al. (2009) study a new bi-objective variant of the orienteering problem where each customer is associated with two different values of profit. The two objective functions considered are the maximization of both kinds of collected profits. The authors propose an ant colony optimization and a variable neighborhood search, hybridized both by a path re-linking method, in order to generate Pareto optimal solutions. More recently, Labadie et al. (2011) have designed an NSGA-II based approach to resolve the bi-criteria version of the TOP. In this last study, the aim is to select the set of customers to be serviced and to build a fixed number m greater than one of tours to cover these customers, so as the total profit is maximized and overall traveled distance is minimized.

1.4.4 Covering Tour Problems

The covering tour (CTP for covering tour problem) generalizes the traveling salesman problem (TSP). It considers a graph G defined as in the CVRP but the set of nodes V is constituted of two complementary subsets V_1 and V_2. The first (V_1)

contains the set of nodes that can be visited and contains some vertices which must be included in the solution. The second set (V_2) encloses nodes that must be covered. In addition to these data, a covering distance L is given. The problem aims to build a tour with a minimal length visiting a subset of nodes from V_1 such that all nodes in V_2 are covered. A node v is said to be covered if and only if there exists at least one node in the tour such that the distance separating it from v is less than L. As for the TSP with profits, the covering tour is clearly identified by Boffey (1995) as a multi-criteria problem.

The maximal covering tour problem is a bi-criteria variant of CTP introduced by Current and Shilling (1994). In this problem, for each node to cover in V_2 is associated a demand and the aim is to build a tour containing exactly p nodes from V_1 (with $p \leq |V_1|$), such that the total demand covered is maximal and the cost (or length) of the tour is minimal. In this variant, a node v is said to be covered if and only if its demand is satisfied by a node in the tour contained in the circle whose center is v and radius is L. Such problems are encountered in mobile service delivery systems such as health care delivery in the rural areas of developing countries and in disaster relief supplies where the aim is to ensure the delivery of a large amount of emergency supplies such as food, water, and medicaments to some center points from which the supplies would be distributed to others disaster zones. For a recent survey on covering problems see Farahani et al. (2012), a subsection is dedicated to the problems already mentioned.

Besides the paper of Current and Shilling (1994) in which a heuristic is proposed to generate an approximation of the Pareto front, another study from Jozefowiez et al. (2007) dedicated to the bi-objective covering problem is also available in the literature. In this last paper, the constraint requiring exactly p nodes in the tour is relaxed and the covering distance imposed in the CTP becomes an objective. The problem studied deals with the minimization of the tour cost and the minimization of the cover. The cover of a solution is defined as the maximal distance between nodes which must be covered (nodes in V_2) and their closest nodes included in the tour. The authors have proposed a two-phase cooperative strategy that combines a multi-objective evolutionary algorithm with a branch-and-cut algorithm initially designed to solve a single-objective covering tour problem. The numerical tests are carried on randomly generated instances and real data (data of the Suhum district, east region of Ghana) and the results are compared to those obtained by a bi-objective exact method based on an ε-constraint approach with a branch-and-cut algorithm.

More recently, Tricoire et al. (2012) have studied the bi-objective covering tour problem with stochastic demands. The two considered criteria, both to minimize, are the total cost (opening cost for distribution centers plus routing cost for a fleet of vehicles) and the expected uncovered demand. The authors assume that depending on the distance, a certain percentage of clients goes from their homes to the nearest distribution centers. To compute solutions of the two-stage stochastic program with recourse, a branch-and-cut technique is used within an ε-constraint algorithm. Computational results on real-world data for rural communities in Senegal show the viability of the approach.

Some other studies have appeared recently in the literature and are dedicated to humanitarian logistics and disaster relief optimization where often several conflicting criteria are to be taken into account. In those studies, one is often faced to the resolution of some variants of the covering tour problem. For instance, the study from Viswanath and Peeta (2003) deals with a multi-commodity maximal covering network design problem for identifying critical routes for earthquake response. The problem is formulated as a two-objective (minimizing the total travel time and maximizing the total population covered) integer programming model that is solved with a branch-and-cut. The search for the critical routes for an origin–destination pair is confined to a limited geographical region to reduce the computational time.

Tzeng et al. (2007) propose a multiple objective relief-distribution model with objectives based on the effectiveness (through the minimization of the total cost and the total travel time) and fairness (by maximizing the minimal satisfaction during the planning period) of the overall distribution system. Results of an empirical study are presented.

Nolz et al. (2010) study a multi-vehicle covering tour problem that consists of routing and placement of tanks of water to cover all beneficiaries rather than being transported directly to them. Two objectives are targeted: the first is related to distances between population and distribution points, and the second is related to cost of the chosen tour.

Vitoriano et al. (2011) add another important aspect when dealing with humanitarian problems that is the reliability of the routes. Hence, they proposed to extend the bi-criteria approach proposed in the previous works dedicated to humanitarian aid distribution problems, by considering a multi-criteria optimization model based upon cost, time, equity, priority, reliability, and security. More specifically, the problem is described through a transport network with pick-up, delivery, or connection nodes and arcs characterized by distance, average velocity and reliability, heterogeneous fleet of vehicles, operation elements such as the global quantity to be distributed and the budget available. The problem is not formulated exactly as a covering tour problem, but the proportion of satisfied demand at a specific node is considered. A goal programming model is presented and applied to the Haiti earthquake that happened in 2010.

1.5 Multi-criteria Path, Flow, and Network Design

In some kinds of extension to the multi-depot case, routing problems aim at finding the paths from some origin positions to destination points. Such examples can be found in supply chain or multi-modal transportation. First, let us consider the case of finding the optimal transit only between two nodes of a network. In the single objective case, this problem is referred to as the shortest path problem and has been intensively studied in the literature. Practical applications such as routing in railways networks often show the necessity to compute the shortest path with respect to several criteria such as traveling time minimization, waiting time, or number of transit points. The interested reader on such problems can see the survey from Skriver (2000).

In more recent researches, Raith and Ehrgott (2009) considered the bi-criteria shortest path problem where two kinds of costs are associated with each link in the network. The aim is to compute a path linking an origin point to a destination point such as both overall total costs are minimal. The authors compare several strategies to resolve the problem on grid, random, and road networks. They deduce that the two-phase method is competitive with other commonly applied approaches to solve the bi-criteria shortest path problem and that the two-phase method works well when combined with both a ranking, a label correcting, and a label setting approach in the second phase. However, the tests show that the label correcting and setting approaches are preferable as they are more stable and, although very efficient on some instances, enumerative near shortest path approach is much time-consuming on others. In the same year, another study from Pinto et al. (2009) was developed for the tri-criterion shortest path problem with two bottleneck objective functions (MinMax, MaxMin for instance) and a cost function. An algorithm able to generate a set of Pareto-optimal paths is proposed and the authors show that bottleneck functions with finite number of values lead to algorithms with polynomial complexity. Then, Pinto and Pascoal (2010) have proposed an improved version of the algorithm appeared in the previous paper. Although both algorithms have the same worst case complexity, the improved version is able to improve the running time on randomly generated benchmark.

Ghoseiri and Nadjari (2010) are also involved in this issue. They present an algorithm based on multi-objective ant colony optimization (MOACO) and propose experimental analyzes on randomly generated instances with two objective costs to minimize. Compared with results of label correcting solutions (the most known efficient algorithm for solving this problem) on the Pareto optimal frontiers, the suggested algorithm produces good quality non-dominated solutions and time saving in computation of large-scale bi-objective shortest path problems.

Reinhardt and Pisinger (2011) also focus on the multi-objective shortest path problems and give a general framework for dominance tests. This is particularly useful to eliminate paths in a dynamic programming framework when using multiple objectives. The authors report results on instances based on the data of a shipping company with several nonadditive criteria such as the time, the number of transfers, the cost, or the probability of reaching the destination.

The studies of Mora et al. (2013) and Tezcaner and Köksalan (2011) deal with military logistics and concern also multi-objective shortest path design. The first is dedicated to solve a path finding problem considering two objectives: maximization of speed and safety. To solve it, three versions of MOACO algorithms, globally identified as hCHAC and dealing with a different number of objectives (two, four, and just one in an aggregated function) are designed. A different parameterization set has been considered in each case. The hCHAC algorithms are tested in several different (and increasingly realistic) scenarios, modeled in a simulator and compared with other MOACOs. Two of them are well-known state-of-the-art MOACOs and the third is a novel multi-objective Greedy approach used as a baseline. The experiments show that most of the hCHAC algorithms outperform the other approaches, yielding at the same time very good military behavior in the tactical sense. Within the hCHAC family, the approach considering two objectives yields the best results overall. The second study by Tezcaner and Köksalan (2011) addresses the route selection problem for unpiloted aircrafts called unmanned air vehicles (UAV). The problem consists of visiting several targets before returning to the base. Determining a good route in such a case may mean to minimize the total distance traveled and maximizing radar detection threat. However, contrary to classical TSP, there is not a single path between any two consecutive nodes but multiple possible paths. Therefore, the problem turns into a combination of an interrelated multi-objective shortest path problem and an multi-objective traveling salesman problem (MOTSP). The authors develop an exact interactive approach to identify the best paths and the best tour of a decision maker under a linear utility function.

Shimamoto et al. (2010) study a problem for which paths have to be found for various origin and destination nodes of the graph. More specifically, they analyze an existing bus network. In this case, there is no product to ship but the model has to consider the passengers' behavior. To do so, it is formulated as a bi-level optimization problem. The upper problem minimizes costs for both passengers (total travel cost) and operators (total operational cost) while the lower problem deals with the transit assignment. An NSGA-II is proposed to solve a study case on demand data from Hiroshima City.

When deliveries have to be made through a supply chain, the aim might be to design the distribution network. Cintron et al. (2010) describe a multiple criteria mixed-integer linear program to determine the optimal configuration of the manufacturing plants, distributors, and customers in a distribution network and to design the flow of products in this system. In other words, for each customer the model chooses the best option for receiving products based on several criteria: profit, lead time, power, credit performance, and distributors' reputation. The options to supply the products are delivery from (1) the regional distribution center (DC), (2) the manufacturing plant, (3) an independent distributor who is supplied from the regional DC, or (4) an independent distributor who is supplied directly from a manufacturing plant. Tests are performed on real data from a consumer goods company and under multiple scenarios to reflect the variability in demand.

Still working on a supply chain within a three-level logistic network, Rajabalipour-Cheshmehgaz et al. (2013) propose to find compromise solutions through a customized Pareto-based multi-objective evolutionary algorithm, NSGA-II. In this study, the levels are some potential suppliers, distributed centers, and consumers with deterministic demands for a period of time. As in Cintron et al. (2010), some flexibility is possible with potential direct shipments from suppliers to consumers. This is motivated here by the option of capacitated facilities (suppliers and distributed centers). So the problems are formulated into four individual logistic network models varying with the flexibility option and/or the capacitated facilities. The main objective is to calculate the status (open or close) of facilities and transportation links in order to minimize the response time to consumers, the transportation cost, and the facility costs, simultaneously and without considering prior knowledge, through the seasonal network (re)design.

Marjani et al. (2012) consider a supply chain in which distribution centers operate as transfer points (cross-docking) to obtain a least storage all along the system. The coordination of cross-docks is then crucial. The authors considered multi-type and time-restricted pickups and deliveries, transshipment possibility among cross-docks and tardiness permission for some pickups. They modeled the distribution planning problem of the cross-docking network through a bi-objective integer programming model minimizing total transportation and holding costs and total tardiness. They also propose a heuristic procedure to construct an initial solution and three frameworks based on variable neighborhood search, tabu search, and simulated annealing, respectively.

Concerning problems dealing with transfer points, a particular case is the multimodal transport, i.e., routes performed by at least two different means of transport. Androutsopoulos and Zografos (2009) study the determination of non-dominated itineraries when paths enhanced with scheduled departures have to be made in a multimodal network with time-dependent travel times. The authors propose to decompose the problem into elementary itinerary subproblems, solved by a dynamic programming algorithm. Si et al. (2011) work on urban multimodal traffic network and study environmental pollution and energy consumption for such a system, in addition to minimizing the total travel time. The multi-criterion system optimization problem also dealt with factors, such as travelers' convenience which influence their behaviors. A bi-level programming model is proposed, in which the multi-objective optimization model is treated as the upper level problem and a combined assignment model to manage to convenience is processed as the lower level problem. The solution algorithms are given through a single numerical example.

Finally, a mixed between the shortest path, the bus routing and the multi-modal problems, is considered by Artigues et al. (2011). They propose several label setting algorithms for computing the itinerary of an individual in urban transportation networks. Mode restrictions are considered under the concept of viable path, modeled by a non-deterministic finite state automaton (NFA). The aim is the minimization of the travel time and of the number of modal transfers. They show that this bi-objective problem is polynomial in both the number of arcs and nodes of the transportation network and the number of states of the NFA. They also propose dominance rules that allow reducing significantly both the CPU times and the number of visited labels for all algorithms. Tests of their algorithms are performed on a realistic urban network and on an expanded graph.

1.6 Classification of the Literature

This section is dedicated to the summary of the bibliographical review on multi-criteria routing problems. As said before, we are only aware of two published surveys on the subject: Boffey (1995) and Jozefowiez et al. (2008b). Thus, the papers listed here are mainly the ones which have been published over the last half-decade.

Real-life routing problems often consist of a large number of different constraints and objectives; this makes difficult their classification into any specific group of VRPs. Several academic studies listed in the previous sections have aspects that relate them to real-life cases since they have included complex constraints and/or objective functions. The classification proposed here is made through four tables (Tables 1.1, 1.2, 1.3, and 1.4), one per main group of routing problems, giving an overview of the published papers which are presented in an ascending chronological order.

In each table, the first column provides the authors and the publication year of the mentioned paper, so that the interested reader can easily refer to the bibliography section. The second column specifies the problem under consideration. Column 3 recalls the objective functions, with minimize and maximize. Finally, the last column indicates the approach used to solve the problem.

Table 1.1 contains the main publications on classical routing problems and their variants, the most investigated in multi-objective optimization concerning time constrained attributes. This group is the largest with 14 papers. Table 1.2 is dedicated to routing problems with depots. Such kinds of problems are less studied, but this is not surprising since the same is also observed in the mono-objective version. Table 1.3 encloses particular routing problems in which all the customers do not need to be visited. Finally, Table 1.4 covers some extra problems encountered in logistics and involving routing decisions.

Across these tables, an interesting feature clearly appears. A number of the latest papers are dedicated to relief/military contexts or are at least related to a service to maximize; these are marked by a double asterisk (**) in front of the author names. This feature is mainly true for routing problems with optional services since they can be naturally closer to such concern, but also for routing with depots. On the contrary, no reference on this kind of issue is quoted in Table 1.1. However, in an interesting paper, Campbell et al. (2008) propose methodologies to deal with two unusual objective functions for a TSP and a CVRP: one that minimizes the maximum arrival time and the other that minimizes the average arrival time. These criteria are very relevant in a disaster relief context. Even if this is not a multi-objective optimization, the aim of the paper is mainly to show how much impact new objective functions could have on the solutions through approaches based on insertion and local search techniques. Results underline the significant improvements in service to population affected by the disaster.

Table 1.1 CVRP/CARP and its time constrained variants

Authors/year	Problem studied	Objective functions	Used approach
Tavakkoli-Moghaddam et al. (2005)	Vehicle routing problem (VRP) with soft time windows and heterogeneous fleet	Minimize fleet cost Minimize total traveled distance Minimize violation of soft time windows penalty	Simulated annealing
Lacomme et al. (2006)	Capacitated arc routing problem (CARP)	Minimize total traveled distance Minimize makespan	NSGA-II based evolutionary algorithm
Ombuki et al. (2006)	VRP with time windows (VRPTW)	Minimize total traveled distance Minimize number of vehicles	Pareto ranking approach Genetic algorithm
Tan et al. (2006)	VRPTW	Minimize total traveled distance Minimize number of vehicles	Pareto approach Hybrid genetic algorithm
Jozefowiez et al. (2009)	CVRP	Minimize total traveled distance Minimize route imbalance	Multi-objective evolutionary algorithm/ parallelization techniques
Norouzi et al. (2009)	Open VRP with competition based on time windows satisfaction	Minimize total traveling cost Maximize total amount of sales	Multi-objective particle swarm optimization
Wang and Li. (2011)	VRPTW	Minimize total traveled distance. Maximize customer's satisfaction	Hybrid genetic algorithm
Chand et al. (2010)	Capacitated VRP	Minimize total traveled distance Minimize number of vehicles	Genetic algorithm Approach non-Pareto
Ghoseiri and Ghannadpour (2010)	VRPTW	Minimize total traveled distance Minimize number of vehicles	Goal programming approach Genetic algorithm
Gupta et al. (2010)	Fuzzy VRPTW	Maximize customer's satisfaction Minimize fleet size Minimize total traveled distance Minimize waiting time	Genetic algorithm

(continued)

Table 1.1 (continued)

Authors/year	Problem studied	Objective functions	Used approach
Muller (2010)	VRP with Soft Time Windows	Minimize total costs (fleet cost+total traveled distance) Minimize the total penalties	Two-stage heuristic: Solomon insertion heuristic+ejection chains 2-Opt and or-opt operators
Braekers et al. (2011)	VRPTW	Minimize total traveled distance Minimize number of vehicles	Two-phase deterministic annealing algorithm
Garcia-Najera and Bullinaria (2011)	VRPTW	Minimize total traveled distance Minimize overall traveling time Minimize number of vehicles	Multi-objective evolutionary algorithm
Tavakkoli-Moghaddam et al. (2011)	VRPTW with competitive time windows	Minimize total traveling time Maximize total amount of sales	Simulated annealing approach

Table 1.2 Vehicle routing with several depots

Authors/year	Problem studied	Objective functions	Used approach
Lau et al. (2009)	Multi-depot VRP (MDVRP)	Minimize total traveled distance Minimize total traveling time	Fuzzy logic guided NSGA-II
**Tavakkoli-Moghaddam et al. (2010)	Location-routing Problem (LRP)	Minimize total cost (setup cost of the facility, fixed and variable depot costs, and routing cost) Maximize total satisfied demand	Multi-objective scatter search
**Coutinho-Rodrigues et al. (2012)	Location of shelters and evacuation path design in disaster relief context	Minimize total traveled distance for primary and backup paths to shelters, the risk on primary paths and at the shelters, total time required to transfer people from shelters to hospitals, number of shelters	Simulation
**Rath and Gutjahr (2011)	LRP	Minimize the fixed costs for depots and vehicles Mininimize operative cost (routing and warehousing) Maximize the covered demand	ε-constraint method Matheuristic (mixed integer program formulation combined to a variable neighborhood search)

1.7 Conclusion

This work aims to survey the literature dedicated to routing problems and focusses mainly on works which appeared after the review made by Jozefowiez et al. (2008b). We classify the studies on main categories of routing problems clearly identified as multi-objective ones. For some works, when it is not easy to make this classification, we try to keep connection on the different variants presented. Therefore, four main categories are proposed: (1) classical routing problems and their variants concerning time constrained attributes, (2) routing problems with depots, (3) routing problems in which all the customers do not need to be visited, and (4) some extra problems encountered in logistics involving routing decisions.

Over the last half decade, one can observe a growing attention to multi-criteria routing problems. This is due to their numerous real applications and there is still much work to do toward both applications and methodologies. Considering the current state of the literature, we recognize at least two emergent and interesting application fields to be more explored: (1) the first concerns routing problems in military, disaster relief and humanitarian logistics which, in our opinion, disserves

Table 1.3 Routing with optional services (Covering tour-routing with profits)

Authors/year	Problem studied	Objective functions	Used approach
**Viswanath and Peeta (2003)	Multi-commodity maximal covering network design problem	Minimize total traveling time Maximize total population covered	Branch-and-cut
Riera-Ledesma and Juan José Salazar-González (2005)	Traveling purchaser (salesman) problem	Minimize travel cost Minimize purchase cost	Mixed integer linear programming model used in a cutting plane algorithm
**Jozefowiez et al. (2007)	Covering tour problem (CTP)	Minimize tour cost Minimize the cover (the maximal walking distance to node in the tour)	Combined evolutionary algorithm/branch-and-cut algorithm
**Tzeng et al. (2007)	Multiple objective relief-distribution problem	Minimize total cost Minimize total travel time Maximize minimal satisfaction	Fuzzy multi-objective programming
Jozefowiez et al. (2008a)	Traveling salesman problem with profits (TSPP)	Minimize tour length Maximize collected profits	Hybrid ejection chain local search/multi-objective evolutionary algorithm
Bérubé et al. (2009)	TSPP	Maximize collected profit Minimize travel costs	ε-constraint method
Schilde et al. (2009)	Orienteering problem (OP) with two kind of profits/nodes	Maximize both kinds of collected profits	Hybrid ant colony system with path relinking Hybrid variable neighborhood search with path relinking method
**Nolz et al. (2010)	Multi-vehicle CTP	Minimize distance between population and distribution points Minimize total cost	Hybrid method based on genetic algorithms, variable neighborhood search and path relinking
Labadie et al. (2011)	Multiple TSPP	Maximize collected profit Minimize travel costs	NSGA-II based evolutionary algorithm
**Vitoriano et al. (2011)	Humanitarian aid distribution problem	Different criteria in terms of cost, time, reliability, security, and fairness	Goal programming approach
**Tricoire et al. (2012)	CTP with stochastic demands	Minimize total cost (opening cost of centers plus routing cost). Minimize the expected uncovered demand	Combined branch-and-cut and ε-constraint algorithm

Table 1.4 Path, flow, and network design

Authors/year	Problem studied	Objective functions	Approach used
Androutsopoulos and Zografos (2009)	Multi-modal paths with time-dependent travel times	Several criteria such as the minimization of the total cost, time, and transfer points	Decomposition method enhanced with a dynamic programming approach
Pinto et al. (2009)	Tri-criterion shortest Path problem (SPP)	Minimize cost and two bottleneck objective functions	Pareto-based labeling procedure
Raith and Ehrgott (2009)	SPP with two cost values/edges	Minimize simultaneously both costs	Comparative study of the different resolution strategies
Cintron et al. (2010)	Supply chain design	Several criteria: profit, lead time, power, credit performance, and distributors' reputation	Simulation
Ghoseiri and Nadjari (2010)	Multi-criteria SPP	Minimize two objectives based on costs	Multi-objective ant colony optimization
Pinto and Pascoal (2010)	Tri-criterion SPP	Minimize cost and two bottleneck objective functions	Improved version of their algorithm from 2009
Shimamoto et al. (2010)	Multiple origin–destination SPP	Minimize costs for both passengers (total travel cost) and operators (total operational cost) in the first, while the second deals with the transit assignment	Bi-level optimization NSGAII-based approach
Artigues et al. (2011)	Multimodal shortest path computation in urban transportation network	Minimize travel time Minimize number of modal transfers	Label setting algorithms
Reinhardt and Pisinger (2011)	Multi-criteria SPP	Comparison of several criteria such as the minimization of the total cost, time, and transfer points	General framework for dominance tests
Si et al. (2011)	Urban multimodal traffic network	Minimize traffic congestion, air pollution, and energy consumption	Bi-level approach

(continued)

Table 1.4 (continued)

Authors/year	Problem studied	Objective functions	Approach used
**Tezcaner and Köksalan (2011)	Multi-criteria SPP	Minimize total distance traveled Maximize radar detection threat	Exact interactive approach
Marjani et al. (2012)	Supply chain, with cross-docking operations	Minimize total transportation and holding costs and total tardiness	Three metaheuristics: variable neighborhood search, tabu search, and simulated annealing.
**Mora et al. (2013)	Multi-criteria SPP	Maximize speed Maximize safety	Multi-objective ant colony optimization
Rajabalipour-Cheshmehgaz et al. (2013)	Supply chain design	Minimize response time to consumers, transportation cost, and facility costs	Pareto-based multi-objective evolutionary algorithm, NSGA-II

more research in the next years; (2) the second research direction can be oriented toward routing problems in logistics related to the service sector, such as for example maintenance and bus routing problems, where a compromise has to be made between routing costs and the quality of the service.

When examining the summary presented in this review, one can see that most of the developed approaches are based on multi-objective genetic algorithms. The reason that such resolution methods are often chosen is, in our opinion, due to their proven performance on previous studies dealing with combinatorial problems and also due to their ease of implementation. Other metaheuristics known to be efficient in solving vehicle routing problems, such as tabu search or large neighborhood search, must be explored in-depth to adapt them efficiently to the multi-criteria case.

References

Alumur S, Kara BY (2007) A new model for the hazardous waste location-routing problem. Comput Oper Res 34(5):1406–1423

Androutsopoulos KN, Zografos KG (2009) Solving the multi-criteria time-dependent routing and scheduling problem in a multimodal fixed scheduled network. Eur J Oper Res 192(1):18–28

Angus D, Woodward C (2009) Multiple objective ant colony optimization. Swarm Intell 3:69–85

Artigues C, Huguet MJ, Gueye F (2011) State-based accelerations and bidirectional search for bi-objective multimodal shortest paths. LAAS Technical report N°11485

Baldacci R, Mingozzi A, Roberti R (2012) Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. Eur J Oper Res 216:1–6

Belenguer JM, Benavent E, Prins C, Prodhon C, Wolfler-Calvo R (2011) A branch-and-cut method for the capacitated location-routing problem. Comput Oper Res 38:931–941

Bérubé JF, Gendreau M, Potvin JY (2009) An exact ε-constraint method for bi-objective combinatorial optimization problems: application to the traveling salesman problem with profits. Eur J Oper Res 194(1):39–50

Boffey B (1995) Multi-objective routing problems. Top 3(2):167–220

Braekers K, Caris A, Janssens GK (2011) A deterministic annealing algorithm for a Bi-objective full truckload vehicle routing problem in drayage operations. Proced Soc Behav Sci 20:344–353

Bräysy O, Gendreau M (2005a) Vehicle routing problem with time windows, part I: route construction and local search algorithms. Transp Sci 39:104–118

Bräysy O, Gendreau M (2005b) Vehicle routing problem with time windows, part II: metaheuristics. Transp Sci 39:119–139

Campbell AM, Vandenbussche D, Hermann W (2008) Routing for relief efforts. Transp Sci 42(2):127–145

Chand P, Mishra BSP, Dehuri S (2010) A multi objective genetic algorithm for solving vehicle routing problem. Int J Info Tech Knowl Mgmt 2:503–506

Chao IM, Golden BL, Wasil EA (1996) The team orienteering problem. Eur J Oper Res 88(3):464–474

Cintron A, Ravindran AR, Ventura JA (2010) Multi-criteria mathematical model for designing the distribution network of a consumer goods company. Comput Ind Eng 58:584–593

Coello Coello CA, Pulido G, Montes E (2005) Current and future research trends in evolutionary multi-objective optimization. Inf Process Evol Algorithm Adv Info Knowl Pro 213–231

Coello Coello CA (2009) Evolutionary multi-objective optimization: some current research trends and topics that remain to be explored. Front Comput Sci China 3(1):18–30

Coutinho-Rodrigues J, Tralhão L, Alçada-Almeida L (2012) Solving a location-routing problem with a multi-objective approach: the design of urban evacuation plans. J Transp Geogr 22:206–218

Current JR, Schilling DA (1994) The median tour and maximal covering problems. Eur J Oper Res 73:114–126

Deb K (2002). Multi-objective optimization using evolutionary algorithms, 2nd edn. Wiley-Interscience Series in Systems and Optimization, New York

Deb K, Agrawal S, Pratap A, Meyarivan T (2000) A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In: Schoenauer M et al (eds) Parallel problem solving from nature (PPSN VI). Lectures notes in computer science, vol 1917. Springer, Berlin, pp 849–858

Duhamel C, Lacomme P, Prins C, Prodhon C (2010) A GRASP × ELS approach for the capacitated location-routing problem. Comput Oper Res 37:1912–1923

Farahani RZ, Asgari N, Heidari N, Hosseininia M, Goh M (2012) Covering problems in facility location: a review Comput Ind Eng 62(1):368–407

Feillet D, Dejax P, Gendreau M (2005) Traveling salesman problem with profits: an overview. Transp Sci 39:188–205

Garcia-Najera A, Bullinaria JA (2011) An improved multi-objective evolutionary algorithm for the vehicle routing problem with time windows. Comput Oper Res 38:287–300

Ghoseiri K, Ghannadpour SF (2010) Multi-objective vehicle routing problem with time windows using goal programming and genetic algorithm. Appl Soft Comput 10:1096–1107

Ghoseiri K, Nadjari B (2010) An ant colony optimization algorithm for the bi-objective shortest path problem. Appl Soft Comput 10(4):1237–1246

Giannikos I (1998) A multi-objective programming model for locating treatment sites and routing hazardous wastes. Eur J Oper Res 104:333–342

Gupta R, Singh B, Pandey D (2010) Multi-objective fuzzy vehicle routing problem: a case study. Int J Contemp Math Sci 5(29):1439–1454

Hoff A, Andersson H, Christiansen M, Hasle G, Løkketangen A (2010) Industrial aspects and literature survey: fleet composition and routing. Comput Oper Res 37:2041–2061

Jozefowiez N, Semet F, Talbi EG (2005) Enhancements of NSGA II and its application to the vehicle routing problem with route balancing. In Talbi EG, Liardet P, Collet P, Lutton E, and

Schoenauer M, Artificial Evolution 2005, International Conference (EA'2005), Volume 3871, Lecture Notes in Computer Science, Springer, pp. 131–142
Jozefowiez N, Semet F, Talbi EG (2007) The bi-objective covering tour problem. Comput Oper Res 34:1929–1942
Jozefowiez N, Glover F, Laguna M (2008a) Multi-objective meta-heuristics for the traveling salesman problem with profits. J Math Modell Algorithm 7(2):177–195
Jozefowiez N, Semet F, Talbi EG (2008b) Multi-objective vehicle routing problems. Eur J Oper Res 189:293–309
Jozefowiez N, Semet F, Talbi EG (2009) An evolutionary algorithm for the vehicle routing problem with route balancing. Eur J Oper Res 195:761–769
Keller CP, Goodchild M (1988) The multi-objective vending problem: a generalization of the travelling salesman problem. Environ Plan B 15:447–460
Labadie N, Prins C, Reghioui M (2008) A memetic algorithm for the vehicle routing problem with time windows. RAIRO-Oper Res 42:415–431
Labadie N, Melechovský J, Prins C (2011) An evolutionary algorithm for the bi-objective multiple traveling salesman problem. In: Proceedings of the international conference on industrial engineering and systems management (IESM 2011), Metz 25–27 May 2011, Frances, pp 371–379. ISBN 978-2-9600532-3-4
Labadie N, Prins C (2012) Vehicle routing nowadays : compact review and emerging problems. In: Mejia G, Velasco N (Ed) Production systems and supply chain management in emerging countries: best practices, Chapter 8, Springer, pp 141–166
Lacomme L, Prins C, Sevaux M (2006) A genetic algorithm for a bi-objective capacitated arc routing problem. Comput Oper Res 33:3473–3493
Laporte G (2009) Fifty years of vehicle routing. Transp Sci 43(4):408–416
Lau HCW, Chan TM, Tsui WT, Chan FTS, Ho GTS, Choy KL (2009) A fuzzy guided multi-objective evolutionary algorithm model for solving transportation problem. Expert Syst Appl 36:8255–8268
Lenstra JK, Rinnooy Kan AHG (1981) Complexity of vehicle routing and scheduling problem. Networks 11:221–227
Lin CKY, Kwok RCW (2006) 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
List G, Mirchandani P (1991) An integrated network/planar multi-objective model for routing and siting for hazardous materials and wastes. Transp Sci 25(2):146–156
Marjani MR, Moattar Husseini SM, Karimi B (2012) Bi-objective heuristics for multi-item freights distribution planning problem in cross-docking networks. Int J Adv Manuf Tech 58(9–12):1201–1216
Martí R, Campos V, Resende MGC, Duarte A (2011) Multi-objective GRASP with path-relinking. AT&T Labs Research Technical Report
Mora AM, Merelo JJ, Castillo PA, Arenas MG (2013) HCHAC: a family of MOACO algorithms for the resolution of the bi-criteria military unit pathfinding problem. Comput Oper Res 40:1524–1551
Müller J (2010) Approximative solutions to the bicriterion vehicle routing problem with time windows. Eur J Oper Res 202:223–231
Nagata Y, Bräysy O, Dullaert W (2010) A penalty-based edge assembly memetic algorithm for the vehicle routing problem with time windows. Comput Oper Res 37:724–737
Nagy G, Salhi S (2007) Location-routing: issues, models and methods. Eur J Oper Res 177(2):649–672
Nolz PC, Doerner KF, Gutjahr WJ, Hartl RF (2010) A biobjective metaheuristic for disaster relief operation planning. In: Dhaenes C, Jourdan L, Coello Coello CA (eds) Advances in multi-objective nature inspired computing. Studies in computational intelligence. Springer, Berlin, pp 157–177
Norouzi N, Tavakkoli-Moghaddam R, Salamatbakhsh A, Alinaghian M (2009) Solving a novel bi-objective open vehicle routing problem in a competitive situation by multi-objective particle swarm optimization. J Appl Oper Res 1(1):15–29

Ombuki B, Ross BJ, Hanshar F (2006) Multi-objective genetic algorithms for vehicle routing problem with time windows. App Intell 24(1):17–30
Park YB, Koelling CP (1986) A solution of vehicle routing problems in a multiple objective environment. Eng Costs Prod Econ 10:121–132
Pinto L, Bornstein C, Maculan N (2009) The tricriterion shortest path problem with at least two bottleneck objective functions. Eur J Oper Res 198:387–391
Pinto L, Pascoal MMB (2010) On algorithms for the tricriteria shortest path problem with two bottleneck objective functions. Comput Oper Res 37:1774–1779
Prins C, Prodhon C, Soriano P, Ruiz A, Wolfler-Calvo R (2007) Solving the capacitated LRP by a cooperative Lagrangean relaxation-granular tabu search heuristic. Transp Sci 41(4):470–483
Prodhon C (2011) A hybrid evolutionary algorithm for the periodic location-routing problem. Eur J Oper Res 210(2):204–212
Rahoual M, Kitoun B, Mabed H, Bachelet V, Benameur F (2001) Multicriteria genetic algorithms for the vehicle routing problem with time windows. In: Proceedings of fourth metaheuristics international conference, pp 527–532
Rajabalipour Cheshmehgaz H, Desa MI, Wibowo A (2013) A flexible three-level logistic network design considering cost and time criteria with a multi-objective evolutionary algorithm. J Intell Manuf 24(2):277–293
Raith A, Ehrgott M (2009) A comparison of solution strategies for biobjective shortest path problems. Comput Oper Res 36:1299–1331
Rath S, Gutjahr WJ (2011) A math-heuristic for the warehouse location–routing problem in disaster relief. Comput Oper Res (in press) doi: 10.1016/j.cor.2011.07.016
Reinhardt LB, Pisinger D (2011) Multi-objective and multi-constrained non-additive shortest path problems. Comput Oper Res 38(3):605–616
Repoussis PP, Tarantilis CD, Ioannou G (2009) Arc-guided evolutionary algorithm for the vehicle routing problem with time windows. IEEE Trans Evol Comput 13:624–647
Riera-Ledesma J, Salazar-González JJ (2005) The biobjective travelling purchaser problem. Eur J Oper Res 160(3):599–613
Schilde M, Doerner KF, Hartl RF, Kiechle G (2009) Metaheuristics for the bi-objective orienteering problem. Swamr Intell 3(3):179–201
Shimamoto H, Murayama N, Fujiwara A, Zhang J (2010) Evaluation of an existing bus network using a transit network optimisation model: a case study of the Hiroshima city bus network. Transportation 37(5):801–823
Si BF, Zhang HY, Zhong M, Yang XB (2011) Multi-criterion system optimization model for urban multimodal traffic network. Sci China Tech Sci 54(4):947–954
Skriver AJV (2000) A classification of bicriterion shortest path (BSP) algorithms. Asia-Pacific J Oper Res 7:199–212
Solomon MM (1987) Algorithms for the vehicle routing and scheduling problems with time window constraints. Oper Res 35(2):254–265
Tan KC, Chew YH, Lee LH (2006) A hybrid multi-objective evolutionary algorithm for solving vehicle routing problem with time windows. Comput Opt Appl 34(1):115–151
Tavakkoli-Moghaddam R, Safaei N, Shariat MA (2005) A multi-criteria vehicle routing problem with soft time windows by simulated annealing. J Indus Eng 1(1):28–36
Tavakkoli-Moghaddam R, Makui A, Mazloomi Z (2010) A new integrated mathematical model for a bi-objective multi-depot location-routing problem solved by a multi-objective scatter search algorithm. J Manuf Syst 9:111–119
Tavakkoli-Moghaddam R, Gazanfari M, Alinaghian M, Salamatbakhsh A, Norouzi N (2011) A new mathematical model for a competitive vehicle routing problem with time windows solved by simulated annealing. J Manuf Syst 30:83–92
Tezcaner D, Köksalan M (2011) An interactive algorithm for multi-objective route planning. J Opt Theor Appl 150:379–394
Toth P, Vigo D (2002) The vehicle routing problem. SIAM, Philadelphia
Tricoire F, Graf A, Gutjahr WJ (2012) The bi-objective stochastic covering tour problem. Comput Oper Res 39:1582–1592

Tzeng GH, Cheng HJ, Huang TD (2007) Multi-objective optimal planning for designing relief delivery systems. Transp Res Part E 43:673–686
Vansteenwegen P, Souffriau W, Oudheusden DV (2011) The orienteering problem: a survey. Eur J Oper Res 209(1):1–10
Vitoriano B, Ortuño MT, Tirado G, Montero J (2011) A multi-criteria optimization model for humanitarian aid distribution. J Glob Optim 51:189–208
Viswanath K, Peeta S (2003) Multicommodity maximal covering network design problem for planning critical routes for earthquake response. J Transp Res Board 1857:1–10
Wang CH, Li CH (2011) Optimization of an established multi-objective delivering problem by an improved hybrid algorithm. Expert Syst Appl 38(4):4361–4367
Wøhlk S (2008) A decade of capacitated arc routing. In: Golden B, Raghavan S, Wasil E (eds) The vehicle routing problem—latest advances and new challenges. Springer, New York, pp 29–48
Zhou A, Qu BY, Zhao SZ (2011) Multi-objective evolutionary algorithms: a survey of the state of the art. Swarm Evol Comput 1:32–49

Chapter 2
Multi-objective Approaches for Design of Assembly Lines

X. Delorme, O. Battaïa and A. Dolgui

Abstract This chapter deals with the use of multi-objective approaches in the field of assembly line design. The design of assembly or transfer lines is a very important industrial problem, which involves various difficult and interconnected optimization problems. A review of the main multi-objective optimization methods used for these problems is presented and discussed. A case study is also described in order to highlight some interesting properties associated with such multi-objective problems.

Keywords Assembly lines · Line balancing · Multi-objective optimization · Design

2.1 Assembly Line Design

Assembly or transfer lines are production systems which are composed of several workstations organized in a serial manner. Each part successively visits each workstation by moving from one workstation to the next thanks to a linear transportation system, for example, a conveyor belt. Serial flow lines have been initially introduced for the production of large amounts of standardized products (mass-production), but are now also used for the production of families of products with low volume.

Assembly lines are intensively used in various industries (e.g., automotive or electronics) and their properties have been described in scientific literature (Nof et al. 1997).

X. Delorme (✉) · O. Battaïa · A. Dolgui
Henri Fayol Institute, CNRS UMR6158, LIMOS, Ecole des Mines de Saint-Etienne,
158 cours Fauriel 42023 Saint-Etienne Cedex 2, France
e-mail: delorme@emse.fr

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*,
Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_2,

2.1.1 Main Steps of Assembly Line Design

In order to design an assembly line, several important steps (see Fig. 2.1) are usually required:

1. Product(s) analysis: the aim of this step is to provide a complete description of the elementary operations to execute in order to obtain the product(s).
2. Process planning: it covers the selection of processes required to obtain the final product(s) and the definition of technological constraints. For instance, a partial order between operations (precedence constraints) is usually defined but various other restrictions have often to be considered. This step requires an accurate understanding of the functional specifications of the products as well as technological conditions for the operations.
3. Line configuration: this step defines the configuration design which implies the choice of the type of assembly line (e.g., pure serial flow line, hybrid flow shop with parallel stations or U-line), the selection of the equipment needed to perform the operations and the solution of a balancing problem, that is, the allocation of operations to workstations. It is imperative to consider all the technological constraints. At this step, a security margin often has to be considered in order to take into account failures, quality problems and also possible slight modifications of the product.
4. Line layout and transport system design: the material handling system is selected and the layout (placement of machines) is chosen. Products flow is analyzed, usually via simulation, to take into account random events and variability in production.
5. Detailed design and line implementation.

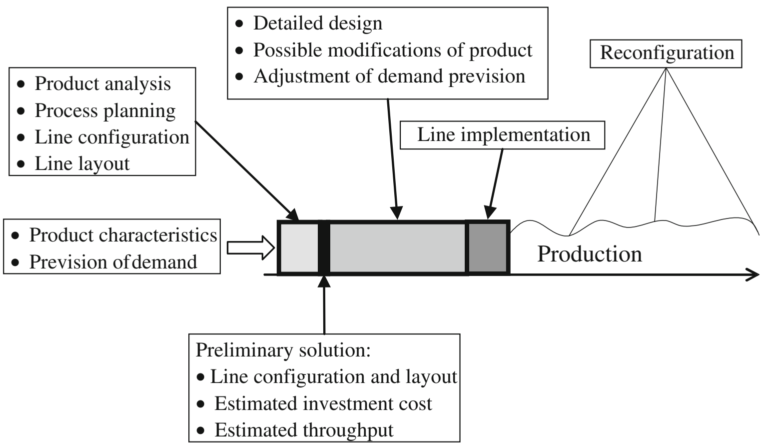


Fig. 2.1 Assembly lines design and reconfigurations

In addition, two other steps can eventually occur after the implementation of the line:

- When the line is designed for the production of several products, a scheduling problem has to be considered in order to determine the sequence of the mix of products.
- When the demand is subject to market fluctuations either in volume or characteristics of the product, the line has to undergo a reconfiguration. A reconfiguration has many similarities with the initial design but the existing line induces specific limitations and objectives.

Considering the complexity of the whole problem, these steps are usually considered sequentially. If the goal of the first two steps is to provide information on the process, the third step corresponds to a combinatorial problem whose objective is to optimize various objectives: minimizing investment costs or future labor costs, maximizing the production rate, minimizing idle times, and smoothing the workload among the workstations.

2.1.2 Line Configuration

Indeed, line configuration is of utmost importance since competitiveness and profitability depend directly on it and this problem has generated a massive amount of scientific publications. Initial studies on this problem have mostly focused on a simple version of the sole balancing problem (SALBP—simple assembly line balancing problem) with only one objective to optimize (number of workstations, production rate, or idle time) and no constraint outside of precedence. Considering the equipment selection and balancing problems independently can make sense, for example, when the equipment selection problem is trivial, either because the resources are interchangeable (e.g., operators without specific competence) or because a particular resource has to be used for each operation (e.g., tools in some lines), but it can be restrictive in many cases. Similarly, most industrial cases involve various technological constraints beside the precedence constraints.

Thus, recent studies have tried to reduce the gap between industrial applications and academic problems. Among the reasons explaining this gap, one of the most important reasons is the difficulty to effectively assess the global performance of a line configuration with a sole criterion. Some studies have considered the criterion of profit maximization to evaluate globally the solutions obtained, but such approaches seem more difficult to apply with increasingly shorter product runs and the growing uncertainty on future demand.

The rest of the chapter is organized as follows. Section 2.2 presents a review of the main multi-objective optimization approaches used for the design of assembly lines such as aggregative and Pareto dominance-based methods. A case study taken from automotive industry is considered in Sect. 2.3. Conclusions and research perspectives are discussed in Sect. 2.4.

2.2 Review of Literature in Multi-objective Assembly Line Optimization

The need to consider multi-objectives in research on assembly line has indeed grown progressively during the last 20 years. Obviously, considering several usually conflicting objective functions to evaluate the quality of feasible solutions instead of a sole one leads to more complex optimization problems.

In this section, we will discuss the main publications from literature on multi-objective decision-making in line balancing. The methods developed in these publications can be split into two main categories: aggregative methods and Pareto-dominance-based methods.

2.2.1 Aggregative Methods

With aggregative methods, authors actually make the assumption that the preferences of the decision-maker can be known a priori. As a consequence, they can define a relationship between the objectives which allows aggregating them. In assembly design, the most commonly used aggregation functions are:

- Lexicographic order, when no compensation is possible between objectives and a priority order can be defined among them.
- Weighted sum, when compensation is fully allowed between objectives with respect to pre-defined substitution rates (i.e., weights).
- Goal programming, when target values (i.e., goals) can be defined for each criterion and the objective is to minimize the deviation from these targets.

2.2.1.1 Lexicographic Order

The first papers using a lexicographic order to aggregate several objective functions for assembly line problems have focused on mixed-model line balancing problems or flexible manufacturing systems. (Gökçen and Erel 1997) studied a problem with three objectives to optimize: the number of workstations, the cycle time, and an objective related to a soft constraint on incompatible zoning. They proposed a preemptive goal programming model inspired from (Deckro and Rangachari 1990) (see Sect. 2.2.1.3) for more details) but their methods actually acts as a lexicographic approach.

In (Sawik 1998), the author considered a line balancing problem combined with parts routing in a flexible assembly system. A two-phase heuristic was proposed to minimize lexicographically the cycle time and the total interstation transfer time.

Pastor et al. (2002) worked on an industrial application corresponding to a multiproduct assembly line balancing problem with four objectives considered in lexicographic order: overall production rate, workload smoothness, cycle time disparity between the different product types, and uniformity of tasks in each workstation. The problem was tackled with two tabu search algorithms which were applied sequentially.

Özcan and Toklu (2009a, b) studied a line balancing problem for the mixed-model two-sided assembly lines. Two performance objectives were considered simultaneously: maximizing the weighted line efficiency and minimizing the weighted smoothness index. They proposed tabu search and simulated annealing algorithms, respectively.

There are also publications on single-model assembly line balancing problems (ALBP). Baykasoğlu (2006) studied two versions of an assembly line balancing problem: straight and U-type line. A multi-rule simulated annealing algorithm was proposed in order to optimize a smoothness index as primary objective and the number of workstation as secondary objective.

The balancing problem associated with U-lines was also investigated by (Gökçen and Ağpak 2006). As in (Gökçen and Erel 1997), the proposed method was inspired from the goal programming model of (Deckro and Rangachari 1990) but was tuned to act as a lexicographic approach. The primary objective to minimize was the number of workstations, with the cycle time as secondary objective and a tertiary objective corresponding to the minimization of the maximal number of tasks assigned per workstation.

Özcan and Toklu (2010) considered two-sided assembly lines with sequence-dependent setup times between tasks. Performing a task directly before another task may influence the latter task inside the same station, because a setup for performing the latter task may be required. Furthermore, if a task is assigned to a station as the last one, then it may cause a setup for performing the first task assigned to that station since the tasks are performed cyclically. A mixed integer program (MIP) was proposed to model and solve the problem. The proposed MIP minimizes the number of mated-stations (i.e., the line length) as the primary objective and it minimizes the number of stations (i.e., the number of operators) as a secondary objective for a given cycle time. A heuristic approach (2-COMSOAL/S) for especially solving large-size problems based on COMSOAL (computer method of sequencing operations for assembly lines) was also presented.

Fattahi et al. (2011) extended the case of two-sided lines to multimanned lines where more than two workers can work at each workstation. A MIP was proposed to solve the balancing problem of the multimanned assembly lines optimally. This model minimizes the total number of workers on the line as the first objective and the number of opened multimanned workstations as the second one. A heuristic based on the ant colony optimization approach was developed to solve the medium- and large-size scales of this problem.

Pastor (2011) introduced a new type of assembly line balancing problem which consists to not only minimize the workload of the most heavily loaded workstation, but then the workload of the second most heavily loaded workstations, then the third, and so on. By nature, such a problem is obviously multi-objective.

Lastly, lexicographic order has also been considered in (Gupta and McGovern 2004) for a disassembly line balancing problem. An ant colony algorithm was proposed which primary sought to minimize the number of workstations, and then to optimize the workload smoothness and the position of hazardous parts in the sequence.

2.2.1.2 Weighted Sum

Contrary to the lexicographic order, the weighted sum was initially used on single-model ALBP. One of the first use of a weighted sum for assembly line balancing was presented by (Leu et al. 1994) as an extension of their work on the use of genetic algorithms to tackle line balancing problems with various objectives. They illustrated their idea with two objectives, namely, workload smoothness and idle time, and suggested to use different weights to obtain several different trade-offs.

In (Ponnambalam et al. 2000), a genetic algorithm was presented for a SALBP. The proposed algorithm sought to optimize a weighted sum of the number of workstations, the line efficiency and a smoothness index. Similarly, Suwannarongsri and Puangdownreong (2009) proposed a tabu search algorithm to optimize a weighted of the same objectives plus the idle time.

Hamta et al. (2011) formulated a flexible task time assembly line balancing problem. Task processing time could be between lower and upper bounds associated with each type of machine available. The machines could compress the processing time of tasks, but this action lead to higher cost. This cost was described in terms of task time via a linear function. A bi-objective nonlinear integer programming model was developed which comprises two inconsistent objective functions: minimizing the cycle time and minimizing the machine total costs. The LP-metric was used to combine these objectives. A genetic algorithm was developed to solve this problem.

Zacharia and Nearchou (2012) presented a fuzzy extension of the SALBP of type 2 with fuzzy task processing times formulated by triangular fuzzy membership functions. The total fuzzy cost function was formulated as the weighted-sum of two bi-objectives fuzzy objectives: (a) minimizing the fuzzy cycle time and the fuzzy smoothness index of the workload of the line; (b) minimizing the fuzzy cycle time of the line and the fuzzy balance delay time of the workstations. A multi-objective genetic algorithm was applied to solve the problem.

Purnomo et al. (2013) considered single model two-sided assembly line problem. The aim of the model was minimizing the cycle time for a given number of mated-workstations and balancing the workstation simultaneously. Genetic algorithm and iterative first-fit rule were used to solve the problem. Based on the experiments, the iterative first-fit rule could take the advantage of finding the best position over many workstations and the genetic algorithm provided more flexible task assignment and was significantly faster than the iterative first-fit rule.

By comparison with single-model problems, the use of weighted sum on mixed-model assembly line problems is rather recent. The only exception corresponds to the work of (Sawik 1997) which considered a weighted sum for his combined balancing and routing problem previously to his work on lexicographic order (Sawik 1998). The main interest of the proposed approach was the interactive procedure with the decision-maker in order to set the value of the weights.

Kara et al. (2007) considered this approach for a combined balancing and sequencing problem in a mixed-model U-line. One objective (workload smoothness) was related to the balancing problem, but the two others (setup cost, smoothness of parts' usage rate) corresponded to the sequencing problem. A simulated annealing algorithm was used in this study. The same combined problem was also studied by (Hwang and Katayama 2010) but with different objectives: two associated with the balancing problem (line efficiency, workload smoothness) and one with the sequencing problem (difference between the actual and average workload).

Hwang and Katayama (2009), proposed an evolutionary approach to deal with workload balancing problems in mixed-model U-shaped lines. The performance objectives considered are the number of workstations and the variation of workload, simultaneously.

Simaria et al. (2009) worked on the same problem of mixed-model two-sided assembly lines than (Özcan and Toklu 2009a) and proposed an ant colony optimisation algorithm, which optimizes a weighted sum of line efficiency and smoothness index.

Kara et al. (2011) studied mixed-model assembly lines with the duplication of common tasks for several models. Three goals relevant to MALB-CD were considered in two pre-emptive goal programming models, one with precise and the other with fuzzy goals, namely minimizing the number of workstations, the cycle time and the total cost required to duplicate common tasks.

Beside these works on deterministic line balancing, (McMullen and Frazier 1998) has proposed a simulated annealing using a weighted sum to deal with a stochastic assembly line balancing problem with parallel stations. The objectives considered in this study were the total labor and equipment cost, workload smoothness, and a probability of lateness.

2.2.1.3 Goal Programming

Goal programming has been one of the first aggregation methods used on multi-objective assembly line problems. Indeed (Deckro and Rangachari 1990) studied a single-model assembly line balancing and proposed an integer linear program with goals to reach for the number of workstations, the cycle time and some soft constraints. Alongside this work, (Malakooti 1991) suggested a goal programming approach for another line balancing problem with the same first two objectives and a third objective corresponding to operating costs instead of the soft constraints.

However, goal programming has actually generated interest in literature by comparison with lexicographic order or weighted sum, and these publications have merely focused on single-model ALBP. More recently, some works have considered a fuzzy extension of goal programming. (Toklu and Özcan 2008) presented a fuzzy goal programming model for the single-model U-line balancing problem with multiple objectives. The first fuzzy goal was the number of workstations in the U-line. The second fuzzy goal was the cycle time. The third fuzzy goal was the maximal number of tasks which were assigned to each workstation in the U-line. A similar approach was used by (Cheshmehgaz et al. 2012) for an assembly line balancing problem with specific objectives related to the ergonomics of the line (posture diversity and accumulated risk posture).

2.2.1.4 Other Aggregative Functions

More anecdotally, some other aggregation methods have been considered for line balancing problems. (Duta et al. 2003) studied a disassembly line balancing problem with two objectives which were aggregated using a ratio function: the outcomes resulting from the valorization of components were divided by the cycle time.

A more sophisticated approach was used in (Gamberini et al. 2006) for a stochastic assembly line reconfiguration problem. In this study, the two objectives considered (labor and incompletion costs, task reassignment) were aggregated using a rank function based on TOPSIS which calculated a distance measure to the ideal and nadir values. Also using a distance measure, (Hamta et al. 2013) worked on an assembly line balancing problem with setup times and operational times varying according to a learning curve. They used a particle swarm optimization algorithm hybridized with a Variable Neighborhood Search to optimize a combination of cycle time, equipment cost and a smoothness index. The main peculiarity of this study was to aggregate these three objectives by using a weighted sum of the distances between the solution considered and a lower bound on each objective.

Finally, some other works could, to some extent, also be considered as multi-objective. For example, the works on SALBP-E deal with two objectives, cycle time and number of workstations, and use the multiplication operator to aggregate both objectives. Another important example of such an implicit aggregation of several objectives comes from profit oriented methods.

2.2.2 Pareto Dominance-based Methods

With Pareto dominance-based methods, authors suppose that the preferences of the decision-maker are unknown. As a consequence, they try to provide a list of interesting trade-offs between the objectives rather than a lone solution. These trade-offs are usually defined by using the Pareto dominance.

Malakooti (1991) was one of the first to study the Pareto front of some single-model ALBP as he tried to deduce some useful properties for a resolution. Following this work, Malakooti and Kumar (1996) proposed a multi-objectives decision support system for ALBP. In their study, they considered five objectives (number of workstations, cycle time, total cost of operations, production rate, and buffer size) but the article actually focused more on interactions with the decision-maker rather than on the optimization problem.

After these works, researches on Pareto-dominance based methods for single-model ALBP have mostly focused on algorithmic methods seeking to approximate the Pareto front. Indeed, in a study of several genetic operators for ALBP with various objectives, Kim et al. (1996) suggested an extension to multi-objective genetic algorithms (MOGA).

Nearchou (2008) considered two versions of the single-model ALBP with two bi-objective: (1) minimizing the cycle time of the assembly line and the balance delay time of the workstations; (2) minimizing the cycle time and the smoothness index of the workload of the line. A new population heuristic was proposed to solve the problem based on the general differential evolution method. The cost function was represented by a weighted-sum of multiple objectives functions with self-adapted weights. The efficiency of the algorithm MODE was compared to a weighted sum Pareto genetic algorithm (GA), and a Pareto-niched GA. The experimental comparisons showed a promising high quality performance for MODE approach. For the second version of the problem, a MODE approach with a new acceptance scheme based on the Pareto dominance concept and a new evaluation scheme based on TOPSIS was proposed by (Nourmohammadi and Zandieh 2011) and a particle swarm optimization algorithm was developed by (Nearchou 2011).

Hwang et al. (2008) presented a MOGA to solve the single-model U-shaped ALBP. The objectives considered were the number of workstations (the line efficiency) and the variation of workload. Chutima and Olanviwatchai (2010) extended the formulation of Hwang and Katayama (2009) by adding a third objective of minimum work relatedness and proposed an evolutionary method with coincidence algorithm. The same problem but with different objectives (cycle time, variation of workload and total operators cost) was studied by (Zhang and Gen 2011) who proposed a generalized Pareto-based scale-independent fitness function genetic algorithm (gp-siffGA) to solve it.

Chica et al. (2010) considered time and space ALBP with the joint minimization of the number and the area of the stations given a fixed cycle time limit and proposed a random greedy search algorithm. Other algorithms were also developed for the same problem: NSGA-II (Chica et al. 2011a) and ant colony optimisation (Chica et al. 2011b). Finally, Chica et al. (2012) developed memetic versions of these algorithms with a multi-objective local search procedure. The memetic advanced NSGA-II showed its excellent performance, obtaining the best solutions.

Similarly, Rekiek et al. (2001) worked on a GA for a line balancing problem with equipment selection. The fitness evaluation of the proposed algorithm was based on the multi-objective decision analysis method Promethee II to order the solutions of the population.

In (Bukchin and Masin 2004), the authors studied a multi-objective ALBP with equipment selection for team oriented assembly systems. The objectives considered were the number of teams, the flowtime (which corresponded to inventory costs) and two objectives related to team oriented assembly systems. They proposed a branch-and-bound algorithm to generate the efficient set as well as some heuristics.

Chen and Ho (2005) considered a specific case of ALBP with equipment selection but without precedence constraints. A MOGA were proposed to obtain potentially efficient solutions for four objectives: total flow time, workload smoothness, cycle time, and tools cost.

Pekin and Azizoglu (2008) addressed the assembly line configuration problem of assigning tasks and equipment to workstations where several equipment alternatives are possible for each task. Minimizing the total equipment cost and the number of work stations were considered together. A branch-and-bound algorithm with powerful reduction and bounding mechanisms was developed.

A multi-objective evolutionary algorithm was also presented in (Shin et al. 2011) with three objectives under consideration: workload smoothness, part movements and tools changes. The approximation of Pareto front obtained by this approach was compared with the solutions from two classic MOGA (NSGA-II and SPEA 2).

Yoosefelahi et al. (2012) considered a robotic ALBP with the following objectives: to minimize the cycle time, robot setup costs and robot costs. Three versions of multi-objective evolution strategies were developed to solve this problem.

Recently, Yang et al. (2013) addressed the reconfiguration problem for a mixed-model assembly line with seasonal demands. The problem was to reassign assembly tasks and operators to candidate stations under the constraint of a given cycle time. The objectives were to minimize the number of stations, workload variation at each station for different models, and rebalancing cost. A MOGA was proposed to solve this problem. A non-dominated ranking method was used to evaluate the fitness of each chromosome. A local search procedure was developed to enhance the search ability of the proposed MOGA.

Chutima and Chimklai (2012) extended the problem of (Özcan and Toklu 2009a) by considering three objectives: (1) to minimize the number of mated-stations, (2) to minimizing the number of workstations or operators, and (3) the tertiary objective consisted of two conflicting sub-objectives to be optimized simultaneously, that is, to maximize work relatedness and minimize workload smoothness. They developed a particle swarm optimization (PSO) algorithm with negative knowledge (PSONK) to solve this problem. In addition, a local search scheme (2-Opt) was embedded into PSONK (called M-PSONK) in order to improved Pareto frontiers obtained.

Another field that has generated some publications corresponds to stochastic ALBP. McMullen and Tarasewich (2006) studied this problem considering four objectives: total cost, probability of lateness, number of workers, and usage rate. They proposed a multi-objective method based on the ACO principles.

Gamberini et al. (2009) considered a stochastic assembly line reconfiguration problem with two joint objectives, total expected completion cost of the new line and similarity between the new and the existing line. A multiple single-pass heuristic algorithm was developed for the purpose of finding the most complete set of nondominated solutions representing the Pareto front of the problem. A multi-objective genetic algorithm was also developed but showed worse results than the heuristic algorithm.

Cakir et al. (2011) dealt with multi-objective optimization of a single-model stochastic ALBP with parallel stations. The objectives were as follows: (1) minimization of the smoothness index and (2) minimization of the design cost. To obtain Pareto-optimal solutions for the problem, an algorithm, based on simulated annealing (SA) was developed. It implemented a multinomial probability mass function approach, tabu list, repair algorithms and a diversification strategy.

Finally, Ding et al. (2010) extended the work of (Gupta and McGovern 2004) on disassembly line balancing by proposing a multi-objective ACO which used an evaluation based on the Pareto-dominance and a niching method.

2.3 Case Study of a Reconfigurable Transfer Line

2.3.1 Problem Description

In this section, we will study a multi-objective problem associated with the design of a reconfigurable machining line (RML). As introduced by (Koren et al. 1999), reconfigurable manufacturing systems are designed to allow easy changes in their physical configuration to answer market fluctuations in both volume and type of product. To achieve this goal, the main required characteristics are: modularity, integrability, customization, convertibility and diagnosability. The use of a RML is motivated by the increasingly shorter product runs and the need for more customization (see Fig. 2.2).

The line under consideration is paced and serial. As for classic assembly lines, the configuration of machining lines corresponds to the assignment of a set of processing operations to workstations which are equipped with a set of machines tools. The usual constraints (processing time of each operation, precedence) must be considered, but machining lines imply several other specific constraints:

- Some subsets of operations must be executed on the same workstation (inclusion constraints);
- Some subsets of operations cannot be executed on the same workstation (exclusion constraints).

Moreover, each workstation is composed of several identical computer numerical controller (CNC) machine-tools to facilitate a future reconfiguration of the line. Within a workstation, each CNC machine executes the same operations (in parallel on different units of products). A part is held at a machine with some fixtures in a given position (part fixing and clamping), but it is possible to rotate the part. However, even after the part rotation or displacement, some sides and elements of the part are not accessible for machining, and the operations which must be processed on these hidden or covered areas cannot be executed (see Fig. 2.3). Therefore, the choice of a part position for part fixing should be also considered in the optimization procedure because it generates specific restrictions for the assignment of operations to workstations.

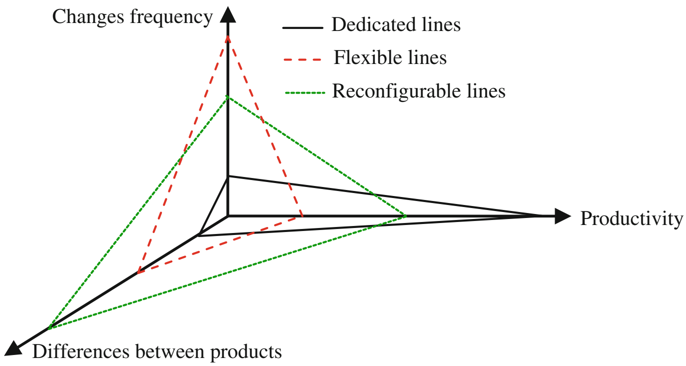
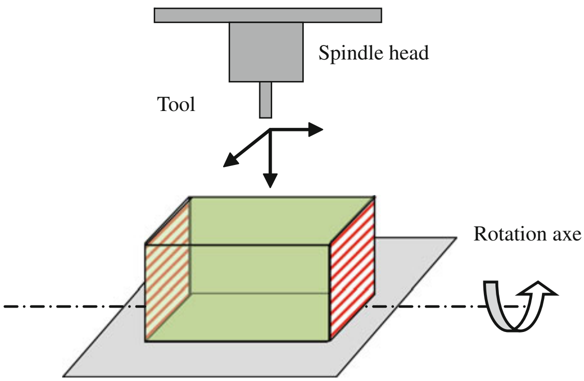


Fig. 2.2 Main differences between dedicated, flexible, and reconfigurable lines

Fig. 2.3 Processing restrictions due to part position



In such a line, all the operations assigned to a station are performed sequentially by the same spindle, and thus sequence-dependent set-up times must be considered. Set-up times are related to the rotation of the part but also change and displacement of the tool. Finally, the sequential execution of operations, as well as the setup times, usually implies large workload times and, as a consequence, parallel machines can be required at some stations in order to increase the production rate. The interest of these lines as well as the main steps of their design has been described in (Delorme et al. 2009).

Now let's introduce the notations used for the various data associated with this problem:

N	set of operations to be assigned;
P_i	set of all predecessors (direct and indirect) of operation i. Precedence constraints define a partial relation of order between operations;
ES	set of subsets $e \in N$ of operations which must be assigned to the same workstation. ES represents the inclusion constraints; that is, the need to carrying out fixed groups of operations on the same workstation;
$\overline{\text{ES}}$	set of pairs of operations (i,j) which cannot be assigned to the same workstation (exclusion constraints); that is, the impossibility of carrying out certain subsets of operations on the same workstation. All pairs (i,j) in $\overline{\text{ES}}$ are defined such that $i < j$
A	set of possible part positions for part fixing in a machining center;
A_i	subset of part positions which allow to process operation $i\,(A_i \subseteq A)$
t_i	operational time to process operation i;
$t_{i,j}$	setup time needed when operation j is processed directly after operation i. The time required for the execution of two sequential operations (i,j) is thus equal to $t_i + t_{i,j} + t_j$;
TMAX	maximal cycle time considered for the line;
CoS	cost associated with the opening of one workstation;
CoM	cost of one CNC machine;
CoMAX	maximal possible investment cost for the line;
$M = \{1, \ldots, \text{MaMAX}\}$	set of possible numbers of machines on a workstation. When several identical CNC machines are installed on the same workstation, the local cycle time of the workstation is equal to the number of parallel machines multiplied by the line cycle time (takt time);
OpMAX	maximal number of operations which can be assigned to a workstation;
StMAX	maximal number of workstations on the line

A solution of this problem, that is, a feasible line configuration, is composed of several decisions:

- The number of workstations and the assignment of each operation to a workstation (balancing problem);
- The sequence of the assigned operations for each workstation (scheduling problem);
- The part fixing position and the number of CNC machines for each workstation (equipment problem).

Moreover, the choice of a solution among the different feasible configurations is actually a multi-objective problem since enterprises seek to minimize investment cost, which depends on the number of workstations and the number of CNC machines, and to maximize the throughput, which is equivalent to minimize the cycle time, at the same time.

The resulting optimization problem is NP-Hard since the sole balancing or scheduling problems are already NP-Hard even with only one objective. A mixed integer programming (MIP) formulation has been proposed for single objective version of this problem corresponding to the minimization of the investment cost with an upper bound on the cycle time of the line (Essafi et al. 2010), but only small size instances could be solved (less than 20 operations). As a consequence, several heuristics have been proposed to deal with this problem (Essafi et al. 2012; Borisovsky et al. 2012).

2.3.2 Some Interesting Properties for a Multi-objective Optimization

Despite the difficulty of this problem, it presents some very interesting properties which can be used during the optimization process. In this chapter, we will focus on three main properties and discuss how they can be used to tackle the multi-objective version of this problem.

2.3.2.1 Given the Sequence of Operations

Let's consider a given sequence of the operations of set N. In this case, the remaining decisions to be made are:

- The points in the sequence of operations where there is a change of workstation, which corresponds to a balancing problem with a strict total order on operations;
- The part fixing position and the number of CNC machines for each workstation.

These remaining decisions correspond to a multi-objective optimization problem which can be formulated with the following MIP:

$$\text{Minimize}T \tag{2.1}$$

$$\text{Minimize CoS} \times \sum_{i \in N, k \in M} x_{i,k} + \text{CoM} \times \sum_{i \in N, k \in M} k \times x_{i,k} \tag{2.2}$$

s.t.

$$\sum_{i \in N, k \in M} x_{i,k} \leq \text{StMAX} \tag{2.3}$$

$$\sum_{k \in M} x_{i,k} \leq 1, \forall i \in N | i < |N| \tag{2.4}$$

$$\sum_{k \in M} x_{n,k} = 1 \tag{2.5}$$

$$\sum_{j \in [i, i+\text{OpMAX}-1], k \in M} x_{j,k} \geq 1, \forall i \in N | i \leq |N| - \text{OpMAX} + 1 \tag{2.6}$$

$$\sum_{l \in [i,j], k \in M} x_{l,k} \geq 1, \forall (i,j) \in \overline{ES} \tag{2.7}$$

$$\sum_{j \in [\min_{i \in e}\{i\}, \max_{i \in e}\{i\}], k \in M} x_{j,k} = 0, \forall e \in ES \tag{2.8}$$

$$\sum_{j \in \left[i, \max\left\{1 \geq \mathrm{i} | \cap_{\lambda \in [\mathrm{i,l}]} \mathrm{A}_\lambda \neq \emptyset\right\}\right], k \in M} x_{j,k} \geq 1, \forall i \in N \tag{2.9}$$

$$\tau_i \leq k \times T + \text{MaMAX} \times \text{TMAX} \times \left(1 - \sum_{k' \in M, k' \leq k} x_{i,k'}\right), \forall i \in N, k \in M \tag{2.10}$$

$$\tau_i \geq t_i, \forall i \in N \tag{2.11}$$

$$\tau_i \geq \tau_{i-1} + t_i + t_{i-1,i} - \sum_{j \in N, |i - \text{OpMAX} \leq j \leq i} t_j \times \sum_{k \in M} x_{i-1,k}, \forall i \in N | i > 1 \tag{2.12}$$

$$x_{i,k} \in \{0, 1\}, \forall i \in N, k \in M \tag{2.13}$$

$$T \geq 0 \tag{2.14}$$

In this model (2.1–2.14), the decision variables $x_{i,k}$ are equal to 1 if operation i is the last operation assigned to a workstation equipped with k CNC machines, τ_i is the workload time accumulated on the current workstation up to operation i, and T correspond to the cycle time of the line. Note that a solution of this model does not contain a decision on the part fixing positions but the constraint (1.9) ensures that there exists at least one possible part position for each workstation so the remaining decision is trivial.

This MIP can be used to obtain new solutions by setting various upper bounds on one of the objective and optimizing the second objective. This method corresponds to an ε-constraint approach. By this way, we can search for solutions in specific areas of the objectives space.

Another way to deal with this multi-objective problem is to use an aggregative function of the objectives. Various aggregative functions can be used for this purpose, for example, one commonly used aggregation function in line balancing is related to the notion of efficiency as in SALBP of type E. Here, maximizing the efficiency of the line corresponds to the minimization of the multiplication of the cost by the cycle time, which is equivalent to the minimization of the cost per unit of product. The corresponding optimization problem can be formulated with the following MIP:

$$\text{Minimize} \sum_{i\in N,k\in M} (\text{CoS} \times \sigma_i + \text{CoM} \times \mu_i) \tag{2.15}$$

s.t.

$$\sigma_i \geq T - \text{TMAX} \times \left(1 - \sum_{k\in M} x_{i,k}\right), \forall i \in N \tag{2.16}$$

$$\mu_i \geq k \times \left(T - \text{TMAX} \times \left(1 - \sum_{k'\in M,\, |k'\geq k} x_{i,k'}\right)\right), \forall i \in N, k \in M \tag{2.17}$$

$$\text{CoS} \times \sum_{i\in N,k\in M} x_{i,k} + \text{CoM} \times \sum_{i\in N,k\in M} k \times x_{i,k} \leq \text{CoMAX} \tag{2.18}$$

$$T \leq \text{TMAX} \tag{2.19}$$

$$\sum_{i\in N,k\in M} x_{i,k} \leq \text{StMAX} \tag{2.20}$$

$$\sum_{k\in M} x_{i,k} \leq 1, \forall i \in N|i<n \tag{2.21}$$

$$\sum_{k\in M} x_{n,k} = 1 \tag{2.22}$$

$$\sum_{j\in[i,i+\text{OpMAX}-1],\, k\in M} x_{j,k} \geq 1, \forall i \in N|i \leq n - \text{OpMAX} + 1 \tag{2.23}$$

$$\sum_{l\in[i,j],\, k\in M} x_{l,k} \geq 1, \forall (i,j) \in \overline{\text{ES}} \tag{2.24}$$

$$\sum_{j\in[\min_{i\in e}\{i\},\max_{i\in e}\{i\}],\, k\in M} x_{j,k} = 0, \forall e \in \text{ES} \tag{2.25}$$

$$\sum_{j\in\left[i,\max\left\{1\geq \text{i}|\cap_{\lambda\in[\text{i,l}]}\text{A}_\lambda\neq\emptyset\right\}\right],\, k\in M} x_{j,k} \geq 1, \forall i \in N \tag{2.26}$$

$$\tau_i \leq k \times T + \text{MaMAX} \times \text{TMAX} \times \left(1 - \sum_{k'\in M,\, |k'\leq k} x_{i,k'}\right), \forall i \in N, k \in M \tag{2.27}$$

$$\tau_i \geq t_i, \forall i \in N \tag{2.28}$$

$$\tau_i \geq \tau_{i-1} + t_i + t_{i-1,i} - \sum\nolimits_{j \in N, |i - \text{OpMAX} \leq j \leq i} t_j \times \sum\nolimits_{k \in M} x_{i-1,k}, \forall i \in N | i > 1 \tag{2.29}$$

$$x_{i,k} \in \{0, 1\}, \forall i \in N, k \in M \tag{2.30}$$

$$\sigma_i, \mu_i \geq 0, \forall i \in N \tag{2.31}$$

$$T \geq 0 \tag{2.32}$$

In the second model (2.15–2.32), two additional decision variables are used: σ_i is equal to the cycle time of the line if operation i is the last operation assigned to a workstation, and μ_i is equal to the cycle time of the line multiplied by the number of CNC machines which equip the current workstation if operation i is the last operation assigned to a workstation. The remainder of the model is similar to the first model (2.1–2.14), save for constraints (2.18) and (2.19) which bound the cost of the line and its cycle time, respectively.

2.3.2.2 Given the Assignment of Operations to Workstations

Let's now consider a given assignment of all operations to workstations. In this case, the remaining decisions to be made are:

- The sequence of operations for each workstation;
- The number of CNC machines for each workstation.

Note that the choice of the part fixing position is trivial as soon as the assignment of operations to workstations is set. Moreover, both decisions to be made can be considered sequentially: whatever number of CNC machines used in a workstation, a nonoptimal sequence of operations can only imply more setup time and thus cannot lead to a solution with a lower cost or cycle time.

The first decision corresponds to a single machine scheduling problem with sequence-dependent setup time and precedence constraints which has to be solved for each workstation. As shown by (Bigras et al. 2008), this problem is equivalent to a time-dependent traveling salesman problem for which various MIP formulations and algorithms have been proposed.

When the workload of each workstation is known, the second decision can be easily obtained with the following procedure:

1. Generate a first solution S_1 by assigning one CNC machine for each workstation;
2. Calculate the cost and the cycle time of S_1. Note that the cycle time is determined by the workstation which has the larger workload; let's denote this workstation as w_1

3. Set index i to 1
4. If there are MaMAX CNC machines on workstation w_i go to step 9
5. Generate solution S_{i+1} by adding one CNC machine on workstation w_i
6. Calculate the cost and the cycle time of S_{i+1}. Determine the workstation which has the larger local cycle time; let's denote this workstation as w_{i+1}
7. Increment index i of 1
8. Go to step 4
9. Set index p to i
10. End

This procedure allows obtaining a set $S = \{S_1, \ldots, S_p\}$ of solutions which are not dominated by each other. Moreover, any other solution with the same assignment of operations to workstations would necessary be dominated by at least one of the solutions of S.

2.3.2.3 Duplicating and Combining of Workstations

Finally, let's consider a solution corresponding to a feasible RML denoted X. The cost and the cycle time of this solution are denoted C(X) and T(X), respectively. Let's also suppose that we would like to find a solution with a lower cycle time than X. In this case, we can actually decide to consider a production system $X^{(2)}$ composed of two identical production lines X working in parallel. The cost of this production system would have be twice the cost of X but its cycle time would be half those of X. Such reasoning can be generalized with any number of parallel production line as soon as the cost of the corresponding production system don't exceed the upper bound:

$$\begin{cases} C\left(X^{(l)}\right) = l \times C(X) \\ T\left(X^{(l)}\right) = \frac{T(X)}{l} \end{cases}, \forall l \in \{2, \ldots, \text{LiMAX}\} | l \times C(X) \leq \text{CoMAX}$$

where LiMAX corresponds to the maximum number of parallel lines.

As a consequence, duplication permits to generate $\frac{\text{CoMAX}}{C(X)} - 1$ new solutions from any solution X. Note that all the solutions generated by this way will have the same efficiency.

Similarly, we can decide to design a production system composed of two different production lines, X and Y, working in parallel. Let's denote the production system resulting from the combination of X and Y as $\langle X + Y \rangle$. The cost and cycle time of this production system can be calculated as follow:

$$\begin{cases} C(\langle X + Y \rangle) = C(X) + C(Y) \\ T(\langle X + Y \rangle) = \frac{T(X) \times T(Y)}{T(X) + T(Y)} \end{cases}$$

We can easily demonstrate that the cycle time of the combined solution is always lower than those of each initial line, which means that the solution generated by combination neither dominates nor is dominated by any of the initial solutions.

2.3.3 Illustration on a Didactic Example

We will now illustrate some of these properties on a didactic example. All the numerical data of the considered case are indicated in Tables 2.1 and 2.2.

Let's consider the sequence of operations used to present the operations in Table 2.1, that is, $\{1, 2, 3, \ldots, 25, 26\}$. Note that this sequence is feasible since it respects all precedence constraints and there is no incompatible operation between any two operations in inclusions.

Using this sequence, we can use the model (2.15–2.32) to obtain a feasible solution of maximal efficiency. Using the solver IBM ILOG CPLEX 12.4 on a computer Intel® Core™ with 2.20 Ghz CPU and 8 Go of RAM, the optimal solution of this problem (for the given sequence) is obtained in less than 3 s. This solution is reported in Table 2.3. It corresponds to a single line with seven workstations and 15 CNC machines.

Knowing this solution, we can use the procedure described in Sect. 2.3.2.2 as a local search. The first step of this procedure is to determine the optimal sequence of operations for each workstation. Considering the very small size of the seven corresponding sequencing problems, their solution can be obtained nearly immediately. The resulting solution is reported in Table 2.4. This solution actually weakly dominates the initial solution in the sense of Pareto since its cycle time is lower.

We can now apply the second step of the procedure in order to obtain a set S of solutions. All the solutions generated are presented in Table 2.5. The last three columns in this table indicate which workstations are equipped with 1, 2, and 3 CNC machines, respectively. For each solution, the workstation which has the largest cycle time, and thus has no idle time, is indicated in bold. Note that the first two solutions (S_1 and S_2) are unfeasible since they don't respect the maximal cycle time (these values are in italic), so we have generated seven non-dominated solutions.

As indicated in Sect. 2.3.2.3, we can obtain additional solutions by duplicating the solutions of the set S. A total of 18 new solutions can indeed be generated (see Table 2.6); however, one of these solutions $\left(S_9^{(3)}\right)$ is unfeasible since its cost exceeds the maximal value (this value is in italic).

Finally, we can also combine some of these solutions. For example, combining the lines S_3 and S_9 would result in a production system with an overall cycle time

Table 2.1 Numerical data of operations for the case study

Operation	Processing time	Direct predecessors	Incompatible operations	Operations on the same station	Possible part positions
1	10	–	–	3	1, 2
2	20	1	–	–	1, 2
3	15	–	–	1	1, 2
4	12	2	6	–	1
5	10	–	–	–	1, 2
6	15	4	4	–	1, 2
7	8	–	–	–	1, 3
8	16	–	–	10,13	2, 3
9	5	–	–	–	2, 3
10	7	–	–	8,13	2, 3
11	10	2	–	–	1, 2
12	4	–	–	–	1, 2
13	8	–	–	8,10	1, 2, 3
14	12	11	–	–	1, 3
15	10	2	–	–	1, 2
16	5	15	–	–	1, 2
17	7	–	18	–	2, 3
18	3	16	17	21	2, 3
19	4	–	–	–	1, 2, 3
20	6	–	–	–	1, 3
21	10	6, 14	22	18	2, 3
22	8	18, 21	21	–	1, 3
23	3	–	–	–	1, 3
24	4	22	–	–	1, 2, 3
25	3	–	–	–	1, 3
26	7	24	–	–	1, 2
Costs	Workstation		$\text{CoS} = 50,000$		
	CNC machine		$\text{CoM} = 200,000$		
Bounds	Cycle time		$\text{TMAX} = 40$		
	Investment cost		$\text{CoMAX} = 10,000,000$		
	Machines per workstation		$\text{MaMAX} = 3$		
	Operations per workstation		$\text{OpMAX} = 10$		
	Workstations per line		$\text{StMAX} = 10$		
	Parallel lines		$\text{LiMAX} = 3$		

of 13.93 and a total cost of 5,500,000. Similarly, we can also combine the solutions duplicated with a solution of set S (but not two solutions duplicated together because the maximal number of lines in parallel would be exceeded).

Obviously, many solutions generated by these different procedures are dominated but we can extract 32 non-dominated solutions. Figure 2.4 presents the corresponding Pareto front as well as the initial solution obtained by the model

Table 2.2 Setup times between operations of the case study

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
1	–	2	3	3	4	3	3	3	4	2	3	3	3	4	4	3	4	3	3	3	3	2	4	4	3	4
2	3	–	4	3	4	4	3	3	5	3	3	4	3	4	5	3	3	3	4	5	4	5	4	3	3	4
3	2	4	–	5	5	3	4	5	4	2	2	3	4	4	4	5	4	2	4	4	3	4	3	4	4	3
4	4	4	5	–	2	5	4	2	4	4	3	4	3	4	4	3	3	3	4	3	3	3	4	2	3	3
5	3	3	3	4	–	3	3	3	4	4	3	3	5	3	3	4	3	4	5	3	3	3	4	5	4	4
6	4	5	3	3	3	–	3	4	5	4	2	2	3	4	4	4	5	4	2	4	4	3	4	3	4	5
7	4	4	4	5	4	4	–	4	5	5	3	4	5	4	2	2	3	4	4	4	5	4	2	4	4	3
8	2	3	3	3	4	4	3	–	2	3	4	3	3	4	2	2	3	4	4	4	5	4	2	4	4	3
9	5	3	3	4	3	4	5	5	–	5	4	4	4	3	5	3	3	4	3	4	5	3	3	3	4	5
10	3	4	3	4	4	3	3	4	3	–	3	4	3	3	4	3	4	4	3	4	2	4	4	3	4	3
11	4	3	4	4	3	3	5	4	3	3	–	4	5	3	4	4	4	5	4	2	4	4	3	4	3	4
12	3	3	3	4	3	3	3	3	4	4	3	–	3	4	2	4	4	3	5	4	2	3	5	3	5	4
13	3	4	5	4	5	4	3	5	4	5	5	4	–	2	4	5	3	4	3	3	3	5	3	3	2	5
14	3	4	5	3	3	3	4	5	4	5	4	3	3	–	4	2	5	4	4	3	4	4	5	5	4	4
15	4	3	4	5	3	3	3	4	5	4	5	4	3	5	–	5	4	3	4	5	4	3	2	4	3	3
16	4	3	4	3	4	4	3	3	3	4	3	3	3	4	4	–	4	5	2	3	4	2	2	3	3	3
17	4	2	4	5	2	2	2	3	4	3	2	5	5	3	5	5	–	2	2	3	4	4	5	2	4	4
18	4	5	2	5	5	5	3	2	2	2	2	5	2	2	4	5	2	–	4	5	3	4	2	2	3	3
19	5	4	2	2	5	2	2	4	3	4	5	5	3	3	2	5	5	4	–	4	5	3	4	5	3	3
20	3	4	2	2	2	3	5	2	4	4	4	4	2	2	2	2	5	5	5	–	3	2	3	2	3	3
21	5	2	2	5	2	3	5	4	4	3	2	4	5	3	2	2	2	5	4	4	–	2	5	4	2	2
22	2	5	2	3	3	4	2	2	2	2	4	5	4	4	5	4	4	4	2	3	5	–	3	4	4	4
23	5	5	5	5	3	2	3	2	2	5	4	4	4	5	5	4	4	4	2	3	2	4	–	4	5	5
24	4	4	2	5	2	4	3	2	4	5	5	3	4	3	4	2	3	5	3	4	5	4	3	–	3	4
25	3	4	2	5	5	5	4	4	4	5	2	4	3	2	3	2	2	4	2	2	3	5	3	4	–	3
26	2	2	4	2	2	4	4	3	4	2	2	2	3	2	2	2	4	4	2	4	2	4	5	5	5	–

Table 2.3 Solution obtained with the model (2.15–2.32)

Station	Sequence of operations	Workload	Number of CNC machines
1	1, 2, 3, 4	68	3
2	5, 6, 7	39	2
3	8, 9, 10, 11, 12, 13	67	3
4	14	12	1
5	15, 16, 17	31	2
6	18, 19, 20, 21	34	2
7	22, 23, 24, 25, 26	38	2
Cycle time: 22.67			
Cost: 3,350,000			

(2.15–2.32). The procedure which has permitted to generate each solution is also indicated and a curve represents the best efficiency value obtained. Logically, the duplication and combination produce solutions with lower cycle time and larger cost, but all the solutions obtained seem to adequately cover the whole Pareto front.

Table 2.4 Solution obtained with optimal sequence in each workstation

Station	Sequence of operations	Workload	Number of CNC machines
1	3, 1, 2, 4	64	3
2	5, 6, 7	39	2
3	8, 9, 12, 11, 10, 13	65	3
4	14	12	1
5	15, 16, 17	31	2
6	18, 19, 20, 21	34	2
7	22, 23, 24, 25, 26	38	2
Cycle time: 21.67			
Cost: 3,350,000			

Table 2.5 Set *S* of solutions obtained with the procedure of Sect. 2.3.2.2

	Cycle time	Cost	Stations with one CNC machine	Stations with two CNC machines	Stations with three CNC machines
S_1	*65*	1,750,000	1, 2, **3**, 4, 5, 6, 7	–	–
S_2	*64*	1,950,000	**1**, 2, 4, 5, 6, 7	3	–
S_3	39	2,150,000	**2**, 4, 5, 6, 7	1, 3	–
S_4	38	2,350,000	4, 5, 6, **7**	1, 2, 3	–
S_5	34	2,550,000	4, 5, **6**	1, 2, 3, 7	–
S_6	32.5	2,750,000	4, 5	1, 2, **3**, 6, 7	–
S_7	32	2,950,000	4, 5	**1**, 2, 6, 7	3
S_8	31	3,150,000	4, **5**	2, 6, 7	1, 3
S_9	21.67	3,350,000	4	2, 5, 6, 7	1, **3**

Table 2.6 Solutions obtained by duplication

	Cycle time	Cost		Cycle time	Cost
$S_1^{(2)}$	32.5	3,500,000	$S_1^{(3)}$	21.67	5,250,000
$S_2^{(2)}$	32	3,900,000	$S_2^{(3)}$	31.33	5,850,000
$S_3^{(2)}$	19.5	4,300,000	$S_3^{(3)}$	13	6,450,000
$S_4^{(2)}$	19	4,700,000	$S_4^{(3)}$	12.67	7,050,000
$S_5^{(2)}$	17	5,100,000	$S_5^{(3)}$	11.33	7,650,000
$S_6^{(2)}$	16.25	5,500,000	$S_6^{(3)}$	10.83	8,250,000
$S_7^{(2)}$	16	5,900,000	$S_7^{(3)}$	10.67	8,850,000
$S_8^{(2)}$	15.5	6,300,000	$S_8^{(3)}$	10.33	9,450,000
$S_9^{(2)}$	10.83	6,700,000	$S_9^{(3)}$	7.22	*10,050,000*

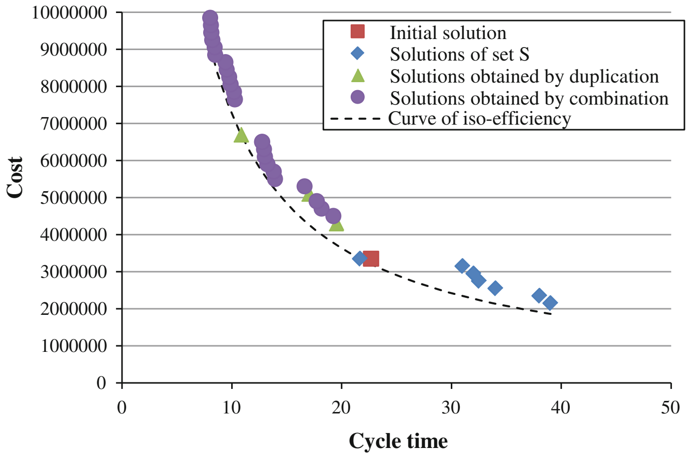


Fig. 2.4 Pareto front generated from the initial sequence

2.4 Conclusion and Perspectives

In this chapter, we have explained the interest to consider multiple criteria in order to support decision-making for the design of assembly lines, and we have presented a review of the main publications in this field.

As highlighted in this review, multi-objective optimization methods for assembly lines have been for a long time really scarce. Authors have mainly focused their research on different versions of the SALBP. The most known multi-objective version of this problem (with cycle time and number of workstations to minimize simultaneously) can be easily tackled with an ε-constraint method. However, this lack of interest is no longer true with the problems studied which involve the most known multi-objective complex constraints and objectives. As a result, publications on multi-objective line balancing problems have been booming in recent years (70 % of the publications referenced in this chapter have been published since 2006), and the number of research articles on this matter should continue to grow. A case study illustrated this trend as there is still a need to develop multi-objective optimization methods that can fully take advantage of the different properties we have described.

Finally, beside the development of efficient optimization methods, the integration of the corresponding models and algorithms within multi-objective decision support systems has not yet generated many contributions, but it is clearly a crucial question for years to come in order to allow a practical use in industry.

References

Baykasoğlu A (2006) Multi-rule multi-objective simulated annealing algorithm for straight and U type assembly line balancing problems. J Intell Manuf 17(2):217–232

Bigras L, Gamache M, Savard G (2008) The time-dependent traveling salesman problem and single machine scheduling problems with sequence dependent setup times. Dis Opt 5(4):685–699

Borisovsky P, Delorme X, Dolgui A (2012) Genetic algorithm for balancing reconfigurable machining lines. Comput Ind Eng. doi:10.1016/j.cie.2012.12.009

Bukchin J, Masin M (2004) Multi-objective design of team oriented assembly systems. Eur J Oper Res 156(2):326–352

Cakir B, Altiparmak F, Dengiz B (2011) Multi-objective optimization of a stochastic assembly line balancing: a hybrid simulated annealing algorithm. Comput Ind Eng 60(3):376–384

Chen JH, Ho SY (2005) A novel approach to production planning of flexible manufacturing systems using an efficient multi-objective genetic algorithm. Int J Mach Tools Manuf 45(7–8):949–957

Cheshmehgaz H, Haron H, Kazemipour F, Desa M (2012) Accumulated risk of body postures in assembly line balancing problem and modeling through a multi-criteria fuzzy-genetic algorithm. Comput Ind Eng 63(2):503–512

Chica M, Cordón O, Damas S (2011a) An advanced multi-objective genetic algorithm design for the time and space assembly line balancing problem. Comput Ind Eng 61(1):103–117

Chica M, Cordón O, Damas S, Bautista J (2011b) Including different kinds of preferences in a multi-objective ant algorithm for time and space assembly line balancing on different Nissan scenarios. Exp Syst Appl 38(1):709–720

Chica M, Cordón O, Damas S, Bautista J (2012) Multi-objective memetic algorithms for time and space assembly line balancing. Eng Appl Artif Intell 25(2):254–273

Chica M, Cordón O, Damas S, Bautista J, Pereira J (2010) Multi-objective constructive heuristics for the 1/3 variant of the time and space assembly line balancing problem: ACO and random greedy search. Inf Sci 180(18):3465–3487

Chutima P, Chimklai P (2012) Multi-objective two-sided mixed-model assembly line balancing using particle swarm optimisation with negative knowledge. Comput Ind Eng 62(1):39–55

Chutima P, Olanviwatchai P (2010) Mixed-model U-shaped assembly line balancing problems with coincidence memetic algorithm. J Softw Eng Appl 3(4):347–363

Deckro R, Rangachari S (1990) A goal approach to assembly line balancing. Comput Oper Res 17(5):509–521

Delorme X, Dolgui A, Essafi M, Linxe L, Poyard D (2009) Machining lines automation. In: Nof DS (ed) Handbook of automation. Springer, New York, pp 599–617

Ding LP, Feng YX, Tan JR, Gao YC (2010) A new multi-objective ant colony algorithm for solving the disassembly line balancing problem. Int J Adv Manuf Technol 48(5–8):761–771

Duta L, Filip F, Henrioud JM (2003) A method for dealing with multi-objective optimization problem of disassembly processes. In: Proceedings of the 5th IEEE intemational symposium on assembly and task planning, Besançon, pp 163–168

Essafi M, Delorme X, Dolgui A (2012) A reactive GRASP and path relinking for balancing reconfigurable transfer lines. Int J Prod Res 50(18):5213–5238

Essafi M, Delorme X, Dolgui A, Guschinskaya O (2010) A MIP approach for balancing transfer line with complex industrial constraints. Comput Ind Eng 58(3):393–400

Fattahi P, Roshani A, Roshani A (2011) A mathematical model and ant colony algorithm for multi-manned assembly line balancing problem. Int J Adv Manuf Technol 53(1–4):363–378

Gamberini R, Grassi A, Rimini B (2006) A new multi-objective heuristic algorithm for solving the stochastic assembly line re-balancing problem. Int J Prod Econ 102(2):226–243

Gamberini R, Grassi E, Regattieri A (2009) A multiple single-pass heuristic algorithm solving the stochastic assembly line rebalancing problem. Int J Prod Res 47(8):2141–2164

Gökçen H, Ağpak K (2006) A goal programming approach to simple U-line balancing problem. Eur J Oper Res 171(2):577–585
Gökçen H, Erel E (1997) A goal programming approach to mixed-model assembly line balancing problem. Int J Prod Econ 48(2):177–185
Gupta S, McGovern S (2004) Multi-objective optimization in disassembly sequencing problems. In: Second world conference on POM and 15th annual POM conference, Cancun
Hamta N, Fatemi Ghomi S, Jolai F, Akbarpour Shirazi M (2013) A hybrid PSO algorithm for a multi-objective assembly line balancing problem with flexible operation times, sequence-dependent setup times and learning effect. Int J Prod Econ 141(1):99–111
Hamta N, Fatemi Ghomi S, Jolai F, Bahalke U (2011) Bi-criteria assembly line balancing by considering flexible operation times. Appl Math Model 35(12):5592–5608
Hwang R, Katayama H (2009) A multi-decision genetic approach for workload balancing of mixed-model U-shaped assembly line systems. Int J Prod Res 47(14):3797–3822
Hwang R, Katayama H (2010) Integrated procedure of balancing and sequencing for mixed-model assembly lines: a multi-objective evolutionary approach. Int J Prod Res 48(21):6417–6441
Hwang R, Katayama H, Gen M (2008) U-shaped assembly line balancing problem with genetic algorithm. Int J Prod Res 46(16):4637–4650
Kara Y, Ozcan U, Peker A (2007) Balancing and sequencing mixed-model just-in-time U-lines with multiple objectives. Appl Math Comp 184(2):566–588
Kara Y, Özgüven C, Seçme N, Chang C (2011) Multi-objective approaches to balance mixed-model assembly lines for model mixes having precedence conflicts and duplicable common tasks. Int J Adv Manuf Techn 52(5–8):725–737
Kim Y, Kim Y, Kim Y (1996) Genetic algorithms for assembly line balancing with various objectives. Comput Ind Eng 30(3):397–409
Koren Y, Heisel U, Jovane F, Moriwaki T, Pritschow G, Ulsoy G et al (1999) Reconfigurable manufacturing systems. CIRP Ann Manuf Technol 48(2):527–540
Leu YY, Matheson L, Rees L (1994) Assembly line balancing using genetic algorithms with heuristic-generated initial populations and multiple evaluation criteria. Dec Sci 25(4):581–606
Malakooti B (1991) A multiple criteria decision making approach for the assembly line balancing problem. Int J Prod Res 29(10):1979–2001
Malakooti B, Kumar A (1996) A knowledge-based system for solving multi-objective assembly line balancing problems. Int J Prod Res 34(9):2533–2552
McMullen P, Frazier G (1998) Using simulated annealing to solve a multi-objective assembly line balancing problem with parallel workstations. Int J Prod Res 36(10):2717–2741
McMullen P, Tarasewich P (2006) Multi-objective assembly line balancing via a modified ant colony optimization technique. Int J Prod Res 44(1):27–42
Nearchou AC (2008) Multi-objective balancing of assembly lines by population heuristics. Int J Prod Res 46(8):2275–2298
Nearchou A (2011) Maximizing production rate and workload smoothing in assembly lines using particle swarm optimization. Int J Prod Econ 129(2):242–250
Nof S, Wilhem W, Warnecke H (1997) Industrial assembly. Chapman Hall, London
Nourmohammadi A, Zandieh M (2011) Assembly line balancing by a new multi-objective differential evolution algorithm based on TOPSIS. Int J Prod Res 49(10):2833–2855
Özcan U, Toklu B (2009a) A tabu search algorithm for two-sided assembly line balancing. Int J Adv Manuf Techn 43(7):822–829
Özcan U, Toklu B (2009b) Balancing of mixed-model two-sided assembly lines. Comput Ind Eng 57(1):217–227
Özcan U, Toklu B (2010) Balancing two-sided assembly lines with sequence-dependent setup times. Int J Prod Res 48(18):5363–5383
Pastor R (2011) LB-ALBP: the lexicographic bottleneck assembly line balancing problem. Int J Prod Res 49(8):2424–2442

Pastor R, Andrés C, Duran A, Pérez M (2002) Tabu search algorithms for an industrial multi-product and multi-objective assembly line balancing problem, with reduction of the task dispersion. J Oper Res Soc 53(12):1317–1323
Pekin N, Azizoglu M (2008) Bi criteria flexible assembly line design problem with equipment decisions. Int J Prod Res 46(22):6323–6343
Ponnambalam S, Aravindan P, Mogileeswar Naidu G (2000) A multi-objective genetic algorithm for solving assembly line balancing problem. Int J Adv Manuf Technol 16:341–352
Purnomo H, Wee H, Rau H (2013) Two-sided assembly lines balancing with assignment restrictions. Math Comp Model 57(1–2):189–199
Rekiek B, De Lit P, Pellichero F, L'Eglise T, Fouda P, Falkenauer E et al (2001) A multiple objective grouping genetic algorithm for assembly line design. J Intell Manuf 12(5–6):467–485
Sawik T (1997) An interactive approach to bicriterion loading of a flexible assembly system. Math Comp Model 25(6):71–83
Sawik T (1998) A lexicographic approach to bi-objective loading of a flexible assembly system. Eur J Oper Res 107(3):656–668
Shin K, Park JO, Kim Y (2011) Multi-objective FMS process planning with various flexibilities using a symbiotic evolutionary algorithm. Comput Oper Res 38:702–712
Simaria A, Zanella de Sá M, Vilarinho P (2009) Meeting demand variation using flexible U-shaped assembly lines. Int J Prod Res 47(14):3937–3955
Suwannarongsri S, Puangdownreong D (2009) Metaheuristic approach to assembly line balancing. WSEAS transactions on systems 2(8):200–209
Toklu B, Özcan U (2008) A fuzzy goal programming model for the simple U-line balancing problem with multiple objectives. Eng Optim 40(3):191–204
Yang C, Gao J, Sun L (2013) A multi-objective genetic algorithm for mixed-model assembly rebalancing. Comput Ind Eng 65(1):109-116
Yoosefelahi A, Aminnayeri M, Mosadegh H, Davari Ardakahi H (2012) Type II robotic assembly line balancing problem: an evolution strategies algorithm for a multi-objective mode. J Manuf Syst 31(2):139–151
Zacharia P, Nearchou A (2012) Multi-objective fuzzy assembly line balancing using genetic algorithms. J Intell Manuf 23(3):615–627
Zhang W, Gen M (2011) An efficient multi-objective genetic algorithm for mixed-model assembly line balancing problem considering demand ratio-based cycle time. J Intell Manuf 22(3):367–378

Chapter 3
Multi-objective Assessment of Warehouse Storage Policies in Logistics and a Fuzzy Information Axiom Approach

E. Çevikcan, İ. U. Sarı and C. Kahraman

Abstract Determining an appropriate storage policy is a critical issue in warehouse management. Storage policies address location assignment of stock keeping units (SKUs) in warehouses. An effective storage policy should not only provide the minimization of transportation and inventory costs, but also increase the level of service available to the internal and external customers. When selecting a storage policy, parameters cannot be frequently determined as crisp values. Fuzzy logic is utilized in many engineering applications so as to handle imprecise data. Moreover, information axiom, the second axiom of axiomatic design (AD), performs the selection of the alternative that mostly satisfies the functional requirements of decision makers. This chapter provides a fuzzy information axiom basis for storage policy selection. After providing background information about storage policies as well as storage assignment models, a fuzzy information axiom-oriented model is introduced. Then, the decision-making model is validated by an application in a company from automotive industry.

Keywords Storage assignment methods · Storage policies · Fuzzy information axiom · Warehouse · Logistics

3.1 Introduction

Warehouses are commercial buildings used for storage of goods by manufacturers, importers, exporter, etc. Some of the warehouse functions are decreasing the expense of transportation costs, serving as a customer service facility, protecting

E. Çevikcan · İ. U. Sarı · C. Kahraman (✉)
Industrial Engineering Department, Istanbul Technical University,
34367 Maçka, İstanbul, Turkey
e-mail: kahramanc@itu.edu.tr

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*,
Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_3,

goods, providing temporary storage of goods, and serving as depots for manufacturing companies.

Storage policies assign items to warehouse storage locations. Items may be assigned randomly, or similar items may be grouped in the same area of the warehouse, or items may be assigned based on order or picking volume. The storage policy is important to the overall system design and budgeting, and the cost of the storage solution can be significant.

A storage assignment method is a set of rules that is used to assign stock keeping units (SKUs) to storage locations. They make decisions about SKU selection and space assignment but do not decide which individual storage locations a SKU should be placed in. The objective of the storage assignment is to evenly balance the expected demand among picking zones. If the number of picks is not balanced across zones, some zones will be idle while other zones are still busy. If the expected demand is evenly distributed, the idle time of zones can be reduced, which matches the lean principle of eliminating idle time (Kong and Masel 2008).

In this study, we make use of a multi-objective decision-making approach to select the best storage policy with respect to the companies requirements. Fuzzy logic is used to determine the linguistic judgments of decision makers for the company requirements. Fuzzy information axiom approach is used to decide the storage policy which meets the company requirements ideally.

The rest of the chapter is organized as follows: storage policies in warehouses are discussed in Sect. 3.2. In Sect. 3.3, a detailed literature review on storage assignment methods is presented. In Sect. 3.4 mathematical programming models for storage assignment are mentioned. In Sect. 3.5 fuzzy axiomatic design is presented and In Sect. 3.6 fuzzy information axiom is applied to define most suitable storage assignment method for a manufacturing firm. Finally, in Sect. 3.7 the chapter concludes with the discussion of findings and future research.

3.2 Storage Policies in Warehouses

A popular approach to reduce the amount of work associated with order picking is to divide the warehouse into a forward area and a reserve area. The forward area is used for efficient order picking. The reserve area holds the bulk storage and is used for replenishing the forward area and for picking the products that are not assigned to the forward area. The forward and reserve area may be distinct areas within the warehouse or the forward and reserve area may be located in the same (pallet) rack. In the latter case, the lower levels represent the forward area, the higher levels represent the reserve area. In some facilities the reserve area is once again subdivided into two separate areas: one for order picking and one for replenishing (Berg and Zijm 1999).

There are numerous ways to assign products to storage locations within the forward and reserve storage areas. The most frequently used storage assignment methods are random storage, closest open location storage, dedicated storage, full turnover storage, class-based storage, duration of stay-based storage, within aisle storage, continuous storage, and shared storage.

3.2.1 Randomized Storage Policy

The simplest storage policy is the random storage policy since it uses no information about the unit-load. It ignores both the product characteristics to which the unit-load belongs or the residence time characteristics of the unit-load. Since no internal structure or partitioning of the storage locations is imposed, the random storage policy requires the smallest possible warehouse size of all storage policies (Goetschalekx 2012). For random storage every incoming pallet (or an amount of similar products) is assigned a location in the warehouse that is selected randomly from all eligible empty locations with equal probability (Petersen 1997). The random assignment method results in a high space utilization (or low space requirement) at the expense of increased travel distance.

3.2.2 Dedicated Storage Policy

In dedicated storage each product is assigned to a fixed location. Replenishments of that product always occur at this same location. A disadvantage of dedicated storage is that a location is reserved even for products that are out of stock. Moreover, for every product sufficient space has to be reserved such that the maximum inventory level can be stored. Thus, the space utilization of this policy is lowest among all storage policies. An advantage is that order pickers become familiar with product locations (Koster et al. 2007). Sometimes, dedicated storage can be useful if products have different weights, i.e., heavy products need to be on the bottom of the pallet and light products on top.

Storage policies based on physical similarity and functional similarity could be defined in dedicated storage policies. In physical similarity based storage policy, items with similar physical characteristics are grouped together in one area. For example, large items are stored in one area, and small items are located in another. This allows the use of similar material handling equipment and similar physical care for each area. In functional similarity based storage policy, functionally related items can be stored together. For example, electrically, hydraulically, and mechanically operated items are grouped in segregated storage areas. The system is especially convenient in manually operated storage facilities in which each warehouse worker becomes knowledgeable in a specific functional area (Sule 1994).

3.2.3 Popularity Based Storage Policies

Every warehouse has items that are retrieved more often than others. In this system these fast-moving items are stored close to receiving and shipping areas, and the slow-moving items are assigned to spaces that are farther away. This arrangement minimizes the distance traveled by warehouse workers in picking orders (Sule 1994). Some storage policies which could be defined in popularity based storage policies are given below:

3.2.3.1 Class-based Storage Policy

The main idea of class-based storage is to divide the available warehouse space into a number of areas. Each item is subsequently assigned to one of the areas, based on the item's demand frequency. Random storage is applied within an area (Roodbergen and Vis 2009). The advantage of this policy is that fast-moving products can be stored close to the depot while the flexibility and high storage space utilization of random storage are applicable (Chan and Chan 2011).

3.2.3.2 Full-turnover-based Storage Assignment Policy

The full-turnover storage policy determines storage locations for loads based on their demand frequency. Frequently requested products get the easiest accessible locations, usually near the input/output points. Slow-moving products are located farther away from the input/output point. An important assumption for this rule is that the turnover frequencies need to be known beforehand (Roodbergen and Vis 2009).

3.2.3.3 Within Aisle Storage Policy

Jarvis and McDowell (l991) presented within aisle storage. In this policy, the highest frequency item is stocked in the first storage location of the first aisle and the second highest one is stocked in the second storage location of the first aisle and so on. After the first aisle is filled, the next highest frequency item is stocked in the first location of the second aisle and so on (Pan and Wu 2012).

3.2.3.4 Across-aisle Policy

In this storage policy, the highest frequency item is assigned to the first location of the first aisle. The next highest frequency one is assigned to the first location of the second aisle and so on. Once the first locations of all the aisles are assigned, the second location of each aisle is then assigned an item. That is, the area that is close to the front aisle contains the high frequency items and the area close to the back aisle contains the low frequency items (Pan and Wu 2012).

3.2.3.5 Reserve Stock Separation-based Storage Policy

In this storage policy, reserve stocks are separated from working stocks. All working stocks are kept together in a compact area from which picking is relatively easy. Reserve stocks from outlying areas replenish the working stocks as the need arises (Sule 1994). There are two types of storage methods depending on reserve stock positions which are floating-slot method and fixed-slot method. In floating-slot storage position method, a SKU is assigned to any vacant storage position. For example, storage position X holds product A. In this method, a new delivery of a product A is received at the warehouse with old product A in the storage position (X). In the storage area, the new delivery of product A is assigned to a second storage position (Y). When product A is required from position X, the computer prints a withdrawal instruction for product A in position X. With the withdrawal of product A from position X, the computer prints next withdrawal instruction for product A from position Y. In fixed-slot method, old product A is allocated to a predetermined number of storage positions. When these positions are depleted of product A, then a new inventory quantity of product A is transferred form the receiving or floating reserve area to replenish the storage positions (Mulcahy 1994).

3.3 Literature Review

Warehouse storage decisions influence almost all key performance indicators of a warehouse such as order picking time and cost, productivity, shipping and inventory accuracy, and storage density (Frazelle 2002). Rouwenhorst et al. (2000), van den Berg (1999),van den Berg and Zijm (1999), Gu et al. (2007, 2010) provide a detailed review of warehouse design, planning, and control models. In this section, the chapters dealing with storage assignment is considered.

Several methods exist for assigning products to storage locations in warehouses. The summary of the reviewed studies is given in Table 3.1. In an early study, Hausman et al. (1976) consider the problem of finding class regions for an AS/RS using the class-based storage assignment method and the single command operating mode. The authors prove that L-shaped class regions where the boundaries of zones accommodating the corresponding classes are square-in-time are optimal with respect to minimizing the mean single command travel time. They also analytically determine optimal storage class-sizes for two product classes. Another early study of storage policy is about the cube-per-order index (COI) rule by Heskett (1963). The COI of an item is defined as the ration of the item's total required space to the number of trips required to satisfy its demand per period. The

Table 3.1 Summary for research on storage assignment policies

Author (year)	Storage assignment policy	Goal	Method
Hausman et al. (1976)	Class-based storage	Finding class regions for an AS/RS	Explicit analytical solutions
van den Berg (1996)	Class-based storage	Minimizing the mean single command cycle time	Dynamic programming
van den Berg and Sharp (1998)	Class-based storage	Minimizing the expected labor time	Binary integer programming
Kulturel et al. (1999)	Turnover rate based	Minimizing average travel time for an AS/RS	Simulation
van den Berg and Gademann (2000)	Randomized, class-based, dedicated storage	Minimizing travel time, response time, and maximizing occupancy rate for an AS/RS	Simulation
Eddy (2004)	Class-based storage, random storage	Minimizing storage cost and retrieval cost	Binary integer programming and simulation
Petersen and Aase (2004)	Randomized, class-based, within aisle storage	Minimizing travel distance	Simulation
Lee and Elsayed (2005)	Dedicated storage	Minimizing total cost	Nonlinear programming
Montulet et al. (1998)	Dedicated storage	Minimizing the peak load	Mixed integer programming
Hassini (2008)	Dedicated storage	Finding the optimal storage quantity	Max–min integer programming
Mupant and Adil (2008a)	Class-based storage	Minimizing order picking and storage space cost	Branch-and-bound algorithm
Muppant and Adil (2008b)	Class-based storage	Minimizing pick travel distance	Branch-and-bound algorithm, dynamic programming
Li et al. (2008)	Class-based storage	Minimizing total travel time	Pareto and niche genetic algorithm
Muppant and Adil (2008c)	Class-based storage	Minimizing storage space cost and order picking cost	Simulated annealing and integer programming
Kovacs (2011)	Class-based storage	Minimizing average picking effort and order cycle time	Mixed integer programming
Chan and Chan (2011)	Randomized, class-based, dedicated storage	Minimizing travel distance and order retrieval time	Simulation

algorithm consists of locating the items with the lowest COI to the locations with the smallest travel time. The reciprocal of the COI is called the turnover rate of that item. Therefore, the COI policy is frequently referred to as the turnover-based storage policy. Goetschalckx and Ratliff (1990) introduce the duration of stay for individual loads as an alternative to the COI. The authors study an ideal situation and remarked that the actual implementation of their approach in real warehouses still needs to be resolved. Rosenblatt and Eynan (1989) present a method for establishing class boundaries for any given number of classes in a square-in-time rack. Eynan and Rosenblatt (1994) extend this method to any rectangular rack. All methods assume a continuous rack and the same demand function as in Hausman et al. (1976). Graves et al. (1977) observe that L-shaped regions are not necessarily optimal when dual commands occur, but they argued that in general they would be no more than 3 % above the optimal. They observe that, in order to enable an incoming load to be stored in its class region, the space requirements increase with the number of classes.

Some attempts have been made to solve storage assignment problem optimally. For example, van den Berg (1996) presents a polynomial time dynamic programming algorithm that distributes products and locations among classes such that the mean single command travel time is minimized. The algorithm allows that the inventory level varies and determines the storage space requirements per class by imposing a risk-level on stock overflow. Van den Berg and Sharp (1998) divide the warehouse into a forward area and a reserve area to reduce the amount of work associated with order picking. They integrated the separation of busy and idle periods to forward-reserve problem in binary integer programming model so as to minimize the expected labor time. In Lee and Elsayed (2005), the problem of the determination of the space requirements for warehouse systems operating under a dedicated storage policy, full turnover-based storage, is investigated. The additional space requirement is satisfied by considering a leased storage space. The warehouse storage capacity problem is then formulated as a nonlinear programming model to minimize the total cost of owned and leased storage space.

Montulet et al. (1998) deals with the problem of minimizing, over a fixed horizon, the peak load in single command cycle dedicated storage policies. Mixed integer programming models are presented and their solutions are compared to turnover-based solutions, which are known to minimize the average load per day. It is stated that turnover-based solutions may not be suitable for the peak load criterion. In Hassini (2008), given a set of storage spaces and a set of products, with specific space requirements and demand rates, the optimal product assignment is obtained. When demand rates are known with certainty, the assignment is found through the solution of a max–min integer program. When demand rates are stochastic with a common law, the assignment is found by solving an integer programming model the objective of which is a non-homogeneous partial difference equation of first order. Muppant and Adil (2008a) address the effects of storage area reduction on order picking and storage space costs are incorporated. A branch-and-bound algorithm is developed to solve the class-based storage model. Muppant and Adil (2008b) utilize branch-and-bound algorithm as well as dynamic programming for forming classes to minimize pick-travel distance. They analyze the relative performance of class-based and dedicated policies to demonstrate that there can be significant savings in using a class-based storage policy. Kovacs (2011) address the problem of storage assignment in a warehouse characterized by multi-command picking and served by milkrun logistics. A mixed integer programming model is proposed for finding a class-based storage policy that minimizes the order cycle time, the average picking effort, or a linear combination of these two objectives.

What is more, in recent years, increasing number of metaheuristic approaches are proposed for storage assignment problem. Li et al. (2008) propose an improved genetic algorithm based on Pareto optimization is designed to solve the storage location assignment problem so as to minimize total travel time. The authors state that the conflicting objectives such as fixed racks stability, picking/storing tasks efficiency, and better customer service should be considered for storage assignment decisions. Park and Seo (2009, 2010) handle planar storage location assignment problem (PSLAP) in shipyards. A mathematical programming model and GA-based (Park and Seo 2009) and dynamic PSLAP heuristic algorithms (Park and Seo 2010) are developed for the solving procedure.

It is demonstrated that the dynamic PSLAP heuristic algorithm performs better than the other solving procedures. Muppant and Adil (2008c) propose a simulated annealing algorithm (SAA) developed to solve an integer programming model for class formation and storage assignment that considers all possible product combinations, storage space cost, and order picking cost. The study shows that SAA gives superior results than the benchmark dynamic programming algorithm for class formation with COI ordering restriction. Chen et al. (2010) address the location assignment and interleaving problem at the same time in an automated storage/retrieval system with duration of stay based shared storage policy. Based on the heuristics for single command operation, a two-step procedure is developed to solve the problem. A tabu search algorithm is proposed to improve the solution for medium- and large-sized problems.

On the other hand, simulation studies are proposed with the aim of comparing different storage assignment policies (Chan and Chan 2011; Kulturel et al. 1999; van den Berg and Gademann 2000; Petersen and Aase 2004). Eddy (2004) develops a binary integer programming model for storage assignment problem with the aim of minimizing storage cost and retrieval cost. Furthermore, the study includes a simulation model developed for the performance analysis of storage policies.

3.4 Mathematical Programming Models for Storage Assignment

In this section, widely known mathematical programming models are included so as to provide an insight of optimization for storage assignment problem.

3.4.1 Binary Integer Programming Model of Askin and Standridge (1993)

A binary integer programming model is proposed for storage assignment problem by Askin and Standridge (1993) with the aim of minimizing travelling cost. In this model, first, warehouse is divided into square grids for allocating products to space. A grid contains one or more storage locations, but all grids have the same storage capacity. Product $i, i = 1, \ldots, N$ requires a maximum of A_i grid squares for storage. The total number of grids is M and the following equation is assumed.

$$\sum_{i=1}^{N} A_i = M \tag{3.1}$$

The warehouse will be allowed to have P shipping and receiving ports. All storage and retrieval requests occur at a port. In fact, the number of loads per time that must pass through each port is known for each product. w_{ip} is proportional to the cost per period for sending product i through port p per unit distance traveled per storage or retrieval request. Normally, w_{ip} will be trips/period. If load transport costs vary for the products, then the w_{ip} should include a factor for cost of product i/unit distance. Thus, the product of w_{ip} with distance/trip will give the total period cost of moving i through port p. Distances will not be known until the assignment of products to grid squares is finalized, but the allocation of storage space into grids allows us to define the parameters d_{pj}—as the distance from the center of grid j to port p. The goal then is to find the set of Ai grids to assign to each product i. This set is denoted as St. Note that if item i is assigned to grid j $(j \in S_i)$, the corresponding travel cost per period due to storage of i in j is c_{ij} where

$$c_{ij} = \frac{1}{A_i} \sum_{p=1}^{P} w_{ip} \times d_{pj} \tag{3.2}$$

Equation (3.2) indicates that $1/A_i$ of product i's flow is to grid j. This expression implies the assumption that all grids for item i use all ports in the same proportion. This may not be true; each port may be served by the closest grid with product i. However, assuming equal port use across a product's grids, we can model the grid assignment problem as a 0–1 program. Let x_{ij} be 1 if product i is assigned to grid j and 0 otherwise. Then,

$$\text{minimize} \sum_{i=1}^{N} \sum_{j=1}^{M} c_{ij} \times x_{ij} \tag{3.3}$$

$$\text{subject to} \sum_{j=1}^{M} x_{ij} = A_i \tag{3.4}$$

$$\sum_{i=1}^{N} x_{ij} = 1 \text{ for all } j \tag{3.5}$$

$$x_{ij} \in \{0, 1\} \tag{3.6}$$

The objective (3.3) accumulates costs as we assign products to grids. Constraints (3.4) guarantee that i is assigned to A_i grids. Constraints (3.5) ensure that each grid is used. Otherwise, the model might naively place several products in the same grid. The generalized assignment problem is a special case of the transportation Problem and can be solved relatively easily. The transportation analogy is that each of the N products is a source. Source i must ship A_i units. Each of the M grid destinations is required to receive one unit.

Although the current model is computationally tractable, Francis and White (1974) point out that one special case makes solution almost trivial. Assume that all products use all ports in the same proportion. Such would be the case, for instance, if all loads enter through one port and leave through another or, of course, if there were only one port. This important factoring assumption can be stated as;

$$w_{ip} = c_i \times w_p \tag{3.7}$$

Ordinarily, c_i will be the total volume of product i moving in and out of storage per time, possibly weighted by a cost per unit distance moved for product i. Then, w_p is the proportion of loads that use port p. Note that w_p must be independent of both product and grid selected. In this special case we have

$$c_{ij} = \frac{1}{A_i}\sum_{p=1}^{P} w_{ip} \times d_{pj} = \frac{c_i}{A_i}\sum_{p=1}^{P} w_p \times d_{pj} \tag{3.8}$$

Letting

$$f_j = \sum_{p=1}^{P} w_p \times d_{pj} \tag{3.9}$$

We see that c_{ij} factors into the product of two terms, one based on the product only and one based on the grid only. The total objective becomes:

$$\sum_{i=1}^{N}\sum_{j \in S_i} c_i \times f_j / A_i \tag{3.10}$$

Each f_j is matched with a c_i/A_i. Now suppose that you are given two sets of numbers and you desire to order the numbers in each set such that the vector product is minimized. The vector product is minimized by matching small values in the first set with large values in the second set, and vice versa. This provides a simple solution algorithm when the factoring assumption holds. The algorithm puts the products with the highest throughput per grid into the lowest cost grids.

Step 1. (order grids) Compute f_j, $j = 1, \ldots, M$ using Eq. (3.9). Place the grids in non-decreasing order of f_j, that is, $\mathrm{f}_{[1]} \leq \mathrm{f}_{[2]} \leq \ldots \leq \mathrm{f}_{[M]}$
Step 2. (order products) Put products in non-increasing order, that is,

$$\frac{c_{[1]}}{A_{[1]}} \geq \frac{c_{[2]}}{A_{[2]}} \geq \ldots \geq \frac{c_{[N]}}{A_{[N]}} \tag{3.11}$$

Step 3. (assign products) For $i = 1, \ldots, N$ assign product $[i]$ to the first $A[i]$ grid squares still available.

3.4.2 *Binary Integer Programming Model of Van den Berg and Sharp (1998)*

Van den Berg and Sharp (1998) focus on operations that observe busy and idle periods. In these operations, it is possible to reduce the number of replenishments in busy periods, by performing replenishments in the preceding idle periods. Prior to the picking period, the forward area is replenished in advance. Their objective is to find an allocation of product quantities to the forward area, which minimizes the expected labor time during the picking period. The following notation is used:

- S: set of products assigned to the forward area,
- P_i: random variable representing the number of picks for product i during the picking period, $i = 1, 2, \ldots, N$,
- R_{ij}: random variable representing the number of concurrent replenishments for product i, if the forward area contains j unit-loads of product i at the beginning of the picking period,
- $i = 1, 2, \ldots, N \quad j = 1, 2, \ldots, m_i$,
- U_i: random variable representing the number of unit-loads of product i that is needed to fulfill demand during the picking period,
- The expected number of picks from the forward area and the reserve area are given by Eqs. (3.12) and (3.13), respectively.

$$\sum_{i \in S} E(P_i) \tag{3.12}$$

$$\sum_{i \notin S} E(P_i) \tag{3.13}$$

Let z_i denote the number of unit-loads of product i that is stored in the forward area at the beginning of the picking period. Accordingly, the expected number of concurrent replenishments is given by Eq. (3.14).

$$\sum_{i \in S} E(R_{iz_i}) \tag{3.14}$$

an expression for $E(R_{iz})$ is derived.

$$E(R_{iz}) = \sum_{k=z+1}^{\infty} (k-z) \times p(U=k) = \sum_{k=z+1}^{\infty} P(U_i \geq k) = E(U_i) - \sum_{k=1}^{z} P(U_i \geq k) \tag{3.15}$$

Subsequently, they formulate the forward-reserve problem as the binary integer programming problem (B-FRP), using the following notation:

m_i: number of unit-loads available of product $i \quad i: 1, 2, \ldots, N$
p_i: $E(Pi)$,
u_i: $E(Ui) - P(Ui)$
u_{ij}: $P(Ui \geq j) \quad i: 1, 2, \ldots, N \quad j: 2, \ldots, m_i$
V: available storage space in the forward area,
T^{pf}: average time for performing one pick from the forward area,
T^{pr}: average time for performing one pick from the reserve area $\left(T^{pr} > T^{pf}\right)$,
T^{cr}: average time for performing one concurrent replenishment.

They define decision variables x_i for $i = 1, \ldots, N$, and y_{ij} for $i = 1, \ldots, N$ $j = 2, \ldots, m_i$
$x_i = 1$if product i is assigned to forward area; 0, otherwise
$y_{ij} = 1$ if the jth unit-load of product i is replenished in advance; 0, otherwise
(B-FRP)

$$\text{minimize} \sum_{i=1}^{N} \left\{ T^{pf} p_i x_i + T^{pr} p_i (1 - x_i) + T^{cr} (u_i x_i - \sum_{j=\infty 2}^{m_i} u_{ij} y_{ij}) \right\} \tag{3.16}$$

subject to

$$\sum_{i=1}^{N} v_i \left(x_i + \sum_{j=2}^{m_i} y_{ij} \right) \leq V \quad i = 1, \ldots, N \tag{3.17}$$

$$y_{12} \leq x_i \quad i = 1, \ldots, N \tag{3.18}$$

$$y_{ij} \leq y_{i(j-1)} \quad i = 1, \ldots, N \quad j = 3, \ldots, m_i \tag{3.19}$$

$$x_i \in \{0, 1\} \quad i = 1, \ldots, N \tag{3.20}$$

$$y_{ij} \in \{0, 1\} \quad i = 1, \ldots, N \quad j = 2, \ldots, m_i \tag{3.21}$$

The objective function follows from Eq. (3.12)–(3.15) after substituting p_i, $u_{i,}$ and u_{ij} and multiplying each term with the corresponding labor time average. Constraint (3.17) stresses that the space occupied by the unit-loads allocated to the forward area may not exceed the available space. The remaining set of constraints (3.18) and (3.19) allows the jth unit-load of product i to be stored in advance, only if unit-loads $1, 2, \ldots, (j-1)$ of product i are assigned to the forward area, for $i = 1, 2, \ldots, N$.

3.4.3 Dynamic Programming Model of Van den Berg (1996)

Van den Berg (1996) presents a polynomial time dynamic programming algorithm that partitions products and locations into classes such that the mean single command cycle time is minimized. The algorithm works under any demand curve, any travel time metric, any warehouse layout, and any positions of the input station and output station. The following notation is used:

Q_i: independent random variables representing the number of unit-loads present of product i at an arbitrary epoch
P_k: set of products in class $k = 1, 2, \ldots, K$

Due to the demand and supply processes the inventory level fluctuates. We estimate the storage space requirement such that the storage space in every class suffices for at least a fraction $0 < \alpha < 1$ of the time. In other words, the probability of a stock overflow is less than $1 - \alpha$. Let Q^k be a random variable representing the inventory level of class k at an arbitrary epoch, i.e.,

$$Q^k = \sum_{i \in P_k} Q_i \tag{3.22}$$

Now, we want to find the smallest size S^k for the class-region of class k such that

$$P(Q^k \leq S^k) \geq \alpha \tag{3.23}$$

Let t_j^{in} denote the travel time between the input station and location j and let t_j^{out} denote the travel time between the output station and location j. Every stored unit-load is retrieved some time later, so that over a long time period half of the single command cycles are storages and half are retrievals. Accordingly, the mean single command cycle time to location $j \in L$, equals:

$$\frac{1}{2}\left(2t_j^{in} + 2t_j^{out}\right) = \left(t_j^{in} + t_j^{out}\right) \tag{3.24}$$

The single command cycle time, $E(SC)$, is defined as

$$E(SC) = \sum_{k=1}^{K} \frac{\sum_{i \in P_k} E(D_i)}{\sum_{i \in P} E(D_i)} \times \sum_{j \in L_k} \frac{(t_j^{in} + t_j^{out})}{|L_k|} \tag{3.25}$$

where, L_k denotes the set of storage locations of class k. The first factor represents the probability that a request concerns class k. The second factor represents the mean travel time to a location in class k. In order to minimize the expected single command cycle time, we assign the products i that constitute the largest demand per reserved space and the locations j with the smallest $\left(t_j^{in} + t_j^{out}\right)$ to the first class and we assign the products i that constitute the next largest demand per reserved space and the locations j with the next smallest $\left(t_j^{in} + t_j^{out}\right)$ to the second class, and so on. Accordingly, the locations are ranked according to nondecreasing $\left(t_j^{in} + t_j^{out}\right)$ and the products are ranked according to non-increasing demand per reserved space. We define $g_k(p, l)$ as the contribution of classes 1, 2,…,k to Eq. (3.24), when products 1, 2,…, p and storage locations 1, 2,…, l are distributed among these classes such that $g_k(p, l)$ is minimal. Then $g_k(p, l)$ satisfies

$$g_k(p,l) = \min_{1 \le p, 1 \le j \le l} \{h_{i+1,p}^{j+1,l} + g_{k-1}(i,j)\} \tag{3.26}$$

where $h_{i+1,p}^{j+1,l}$ denotes the contribution to Eq. (3.23) if the products $i = 1, 2\ldots, p$ and the locations $j = 1, 2,.., l$ form one class k. Recalling that the number of locations required in each class is determined by Eq. (3.22), the values $g_k(p, l)$ are found by iteratively solving the dynamic programming Eq. (3.25). Each $g_k(p, l)$ corresponds to an optimal solution of the subproblem with k classes and the first p products and the first l storage locations when ranked as indicated before.

3.4.4 *Binary Integer Programming Models of Bartholdi and Hackman (2008)*

3.4.4.1 Model for Dead-Heading Minimization

The movement of forklifts or other unit-load equipment is useful if a pallet is being moved; but it does not add value if the forklift is dead-heading (traveling with empty forks). For example, a forklift will deadhead to a storage location to get a pallet, but then travel productively in carrying that pallet to shipping. One way to reduce labor is to store product in convenient locations so that travel with the pallet is decreased. Another way is to reduce deadheading by careful interleaving of put-aways and retrievals, so that after a put-away, the forklift travels directly to pick up another pallet. For the dual-cycle protocol to be most effective, stows and retrievals should be paired to minimize dead-heading. For a given set of planned stows and retrievals the problem of finding an effective pairing can be modeled as a binary integer programming model. This can be done in an ad hoc manner simply by assigning stows to nearby retrievals.

Assume there is a task list consisting of stows $i = 1,\ldots, m$ and retrievals $j = 1,\ldots, n$. Let the shortest distance between the location of stow i and the location of retrieval j be d_{ij}. The shortest distances from each stowage location to all the retrieval locations must be computed, for example, by repeated use of the Shortest Path Algorithm (Bartholdi and Hackman 2008).

Then the problem of finding the pairings of stows and retrievals to minimize total dead-heading may be expressed mathematically in the following mathematical programming model where $x_{ij} = 1$ indicates that the forklift making stow i should then proceed in the most direct way possible to retrieval j.

$$\text{minimize} \sum_{i=1}^{n} \sum_{j=1}^{m} d_{ij} \times x_{ij} \tag{3.27}$$

subject to

$$\sum_{j=1}^{M} x_{ij} = 1 \quad \text{for all } i \tag{3.28}$$

$$\sum_{i=1}^{N} x_{ij} = 1 \quad \text{for all } j \tag{3.29}$$

$$x_{ij} \in \{0, 1\} \tag{3.30}$$

The first constraint requires that each stow be paired with some retrieval, and the second constraint that every retrieval be paired with some stow. If there are fewer stows than retrievals, then we simply add enough "dummy" stows to make them equal, where each dummy stow represents travel from the shipping dock to a retrieval location. Similarly, if there are more stows than retrievals, we add dummy retrievals, each of which represents dead-heading to the receiving dock.

3.4.4.2 Model for Forward Pick Area Allocation

The main insight in this sub section is that once one has decided to store a product in the forward pick area, giving it additional storage locations, beyond the minimum required, conveys no benefit. It does not increase the number of picks from the forward area, nor does it reduce the number of restocks. There is additional savings only when one puts every pallet of the SKU in the forward area so that no restocking is required (that is, no internal moves from bulk storage to the forward pick area). Therefore, the only amounts to consider storing are: no pallets, the minimum practical number of locations, or else all the pallets. This can be formalized as follows:

Let p_i be the number of picks for less than pallet quantities, d_i the number of pallets moved by such picks, and D_i the number of pallets moved by full-pallet picks. Let l_i be the minimum number of locations required by SKU i in the fast-pick area and and u_i be the maximum number of forward locations. (This value might be guessed from historical records; but unless there is some confidence in the upper bound, it is best taken as $u_i = 1$.) Suppose that, on average, it saves s minutes when a pick is made from the forward area rather than from bulk storage; and that each restock of the forward area (that is, each move of a pallet from reserve to the forward area) requires cr minutes. Then, the net benefit of allocating x forward locations to SKU i is

$$net\, benefit = \left\{ \begin{array}{cc} 0 & if\; x = 0 \\ sp_i - c_r d_i & if\; l_i < x < u_i \\ s(p_i + D_i) & if\; x = u_i \end{array} \right\} \tag{3.31}$$

Notice that some SKUs could positively hurt efficiency if they were stored in the forward pick area in less than their maximum amounts. For such a SKU i the net benefit is negative if

$$\left(\frac{p_i}{d_i}\right) s < c_r \tag{3.32}$$

On the other hand, for any SKU, the net benefit of storing all of its pallets in the forward pick area is always positive because no restocking from bulk storage is required. The difficulty here is to know how many constitute "all".

Assume for now that "all" is quite large in relation to the size of the forward pick area and so we take $u_i = \infty$. We can write the problem of selecting SKUs for forward storage by means of choice variable $x_{ij} \in \{0, 1\}$.

$$\text{minimize} \sum_{i=1}^{n} (c_1 p_i + c_r d_i) x_i + c_2 p_i (1 - x_i) \tag{3.33}$$

$$\text{subject to} \sum_{i=1}^{n} l_i x_i \le N \tag{3.34}$$

$$x_i \in \{0, 1\} \tag{3.35}$$

The object is to minimize total labor costs (picking plus restocking) subject to the space constraint that only N storage locations are available in the forward pick area. Let c_1 be the average cost per pick from the forward area and let c_2 be the average cost per pick from bulk storage. We may assume $c_1 < c_2$.

3.5 Fuzzy Axiomatic Design

The most important concept in AD is the existence of the design axioms. The first design axiom is known as the independence axiom and the second axiom is known as the information axiom (IA). They are stated as follows (Suh 1990).

Axiom 3.1 The Independence Axiom: Maintain the independence of functional requirements

Axiom 3.2 The IA: Minimize the information content

The independence axiom states that the independence of functional requirements (FRs) must always be maintained where FRs are defined as the minimum set of independent requirements that characterizes the design goals. The IA states that the design with the smallest information content among those satisfying the first axiom is the best design (Suh 2001).

3.5.1 Crisp Information Axiom

Information is defined in terms of the information content, I, that is related in its simplest form to the probability of satisfying the given FRs. Information content I_i for a given FR_i is defined as follows:

$$\mathrm{I}_i = \log_2\left(\frac{1}{p_i}\right) \tag{3.36}$$

where p_i is the probability of achieving the functional requirement FR_i and log is either the logarithm in base 2 (with the unit of bits). This definition of information follows the definition of Shannon (1948), although there are operational differences. Because there are n FRs, the total information content is the sum of all these probabilities. If I_i approaches infinity, the system will never work. When all probabilities are one, the information content is zero, and conversely, the information required is infinite when one or more probabilities are equal to zero (Suh 1995).

In any design situation, the probability of success is given by what designer wishes to achieve in terms of tolerance (i.e., design range) and what the system is capable of delivering (i.e., system range). As shown in Fig. 3.1, the overlap between the designer-specified "design range" and the system capability range

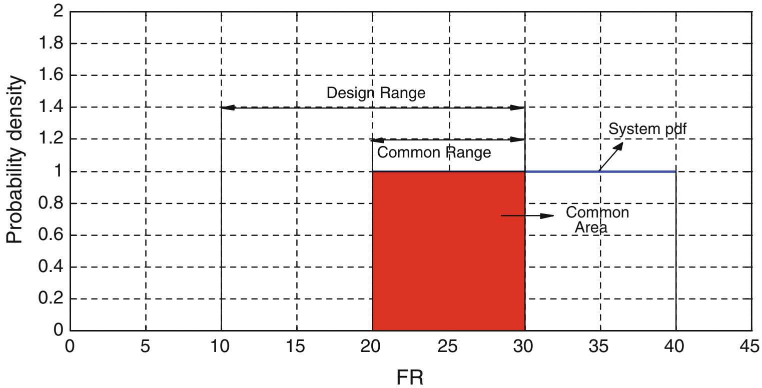


Fig. 3.1 Design range, system range, common range, and probability density function of a FR

"system range" is the region where the acceptable solution exists. Therefore, in the case of uniform probability distribution function p_i may be written as

$$p_i = \left(\frac{\text{Common range}}{\text{System range}}\right) \tag{3.37}$$

Therefore, the information content is equal to

$$I_i = \log_2\left(\frac{\text{System range}}{\text{Common range}}\right) \tag{3.38}$$

The probability of achieving FR_i in the design range may be expressed, if FR_i is a continuous random variable, as

$$p_i = \int_{dr^1}^{dr^u} p_s(FR).dFR \tag{3.39}$$

where p_s(FR) is the system pdf (probability density function) for FR. Equation (3.38) gives the probability of success by integrating the system pdf over the entire design range. (i.e., the lower bound of design range, dr^1, to the upper bound of the design range, dr^u). In Fig. 3.2, the area of the common range (A_{cr}) is equal to the probability of success P (Suh 1990).

Therefore, the information content is equal to

$$I = \log_2\left(\frac{1}{A_{cr}}\right) \tag{3.40}$$

The information content in Eq. (3.40) is a kind of entropy that measures uncertainty. There are some other measures of information in terms of uncertainty.

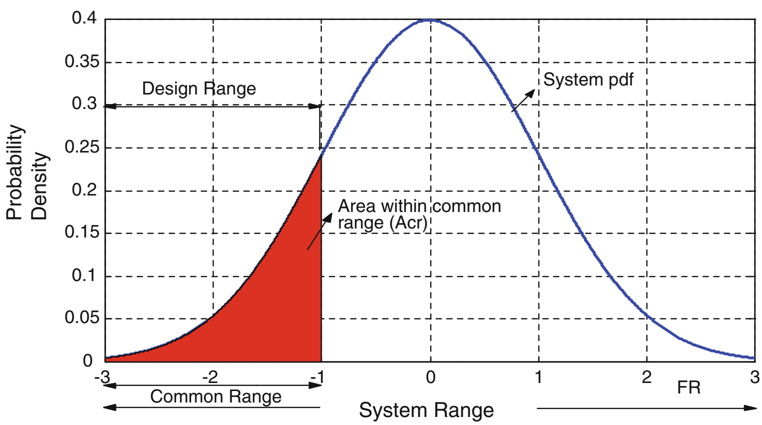


Fig. 3.2 Design range, system range, common range, and probability density function of a FR

Prior to the theory of fuzzy sets, two principal measures of uncertainty were recognized. One of them, proposed by Hartley (1928), is based solely on the classical set theory. The other, introduced by Shannon (1948), is formulated in terms of probability theory. Both of these measures pertain to some aspects of ambiguity, as opposed to vagueness or fuzziness. Both Hartley and Shannon introduced their measures for the purpose of measuring information in terms of uncertainty. Therefore, these measures are often referred to as measures of information. The measure invented by Shannon is referred to as the Shannon Entropy.

The Shannon Entropy, which is a measure of uncertainty and information formulated in terms of probability theory, is expressed by the function

$$H(p(x)/x \in X) = -\sum_{x \in X} p(x) \log_2 p(x) \tag{3.41}$$

where $(p(x)/x \in X)$ is a probability distribution on a finite set X.

Suh's entropy in AD does not require that the total of probabilities is equal to 1.0 while Shannon entropy does. Because of this property, Shannon entropy should not be used as an entropy measure while evaluating independent functional requirements in AD.

3.5.2 Fuzzy Information Axiom Approach

The multi-attribute crisp information axiom approach mentioned above can be used for the solution of decision-making problems under certainty. This approach cannot be used with incomplete information, since the expression of decision variables by crisp numbers would be ill defined. For this reason, multi-attribute fuzzy information axiom is developed in this study. At the same time, a problem including both crisp and fuzzy objectives can be solved by integrating crisp and fuzzy information axiom approaches. This feature is an important advantage which cannot be found in other multi-attribute approaches. The definition and formulation of the developed fuzzy approach are given in the following.

The data relevant to the objectives under incomplete information can be expressed as fuzzy data. The fuzzy data can be linguistic terms, fuzzy sets, or fuzzy numbers. If the fuzzy data are linguistic terms, they are transformed into fuzzy numbers first. Then all the fuzzy numbers (or fuzzy sets) are assigned crisp scores. The following numerical approximation systems are proposed to systematically convert linguistic terms to their corresponding fuzzy numbers. The system contains five conversion scales (Figs. 3.3 and 3.4).

In the fuzzy case, we have incomplete information about the system and design ranges. The system and design range for a certain criterion will be expressed by using 'over a number', 'around a number', or 'between two numbers'. Triangular or trapezoidal fuzzy numbers can represent these kinds of expressions. We now

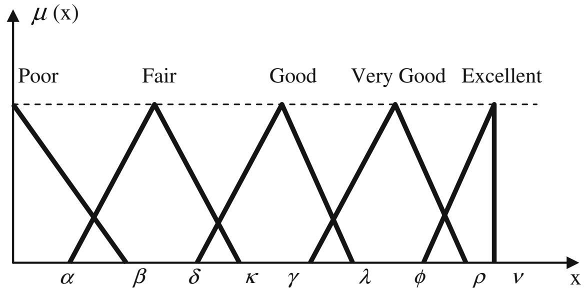


Fig. 3.3 The numerical approximation system for intangible factors

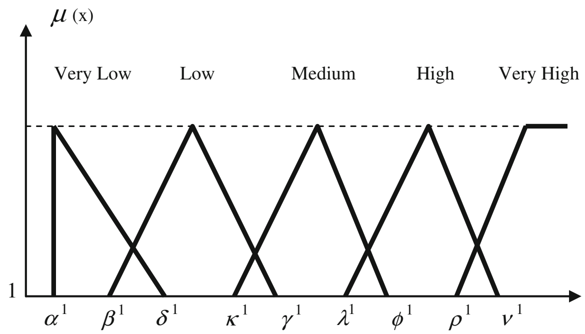


Fig. 3.4 The numerical approximation system for tangible factors

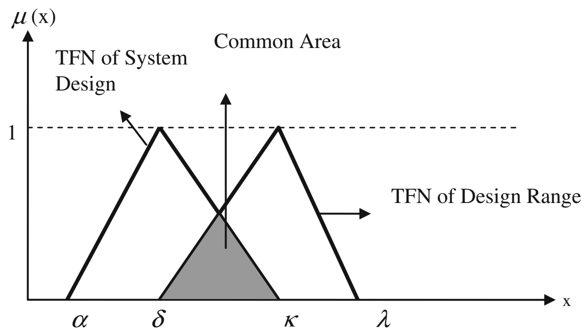


Fig. 3.5 The common area of system and design ranges

have a membership function of triangular or trapezoidal fuzzy number whereas we have a probability density function in the crisp case. So, the common area is the intersection area of triangular or trapezoidal fuzzy numbers. The common area between design range and system range is shown in Fig. 3.5.

Therefore, information content is equal to Eq. (3.42).

$$I = \log_2\left(\frac{\text{TFN of System Design}}{\text{Common Area}}\right) \quad (3.42)$$

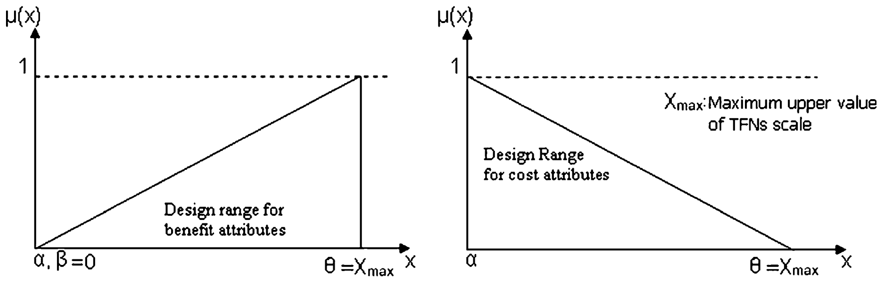


Fig. 3.6 Ideal design ranges for benefit and cost attributes, respectively (Cebi and Kahraman, 2010)

In the basis of IA method, the most important factor is the definition of FRs which is the minimum sets of independent requirements that characterize the design goals. The definition of FRs characterizes the type of the problem and affects the solution of the problem. There are three types of problems; (i) Exact value problems: When the decision maker/makers is/are interested in the alternatives in a specific interval [a, b], the alternatives are eliminated automatically if the values of alternatives are out of this range. (ii) Expected value problems:

Let us assume that decision maker/makers is/are interested in the alternatives in interval [a, b] and another alternative is interval [b, c] which is better than interval [a, b]. However, the effect of the alternative in interval [b, c] is thought to be equal to the effect of the alternative in interval [a, b]. The main advantage of this is that removing inessential advantages of the alternative are omitted since it does not have any importance for the decision maker's aim. Otherwise, it causes the value of the alternative to increase according to the other alternatives. (iii) Ranking problems: IA does not give permission to rank alternatives as in the TOPSIS under the following situations; (1) if there are more than one alternative which does not satisfy the decision goal; (2) if there are more than one alternative satisfying the FRs for all objectives completely. To rank the alternatives, the limits of FRs can be chosen for benefit attributes for $a = 0,\ l(a) = 0$ and for $b = h =$ Xmax (maximum upper value of the alternative in the problem), $l(h) = 1$ and for cost attributes for $a = b = 0,\ l(a) = 1$ and for $h =$ Xmax, $l(h) = 0$. This area is named as ideal FR (IFR) as in Fig. 3.6 (Cebi and Kahraman 2010).

3.6 Application

A company in automotive industry plans to select the best storage policy which satisfies company's objectives and to design its warehouse by applying the selected storage policy. For this purpose experts determine the functional requirements for the company. Six functional requirements are determined as follows:

FR1: Time performance in picking load when processing an order
FR2: Response in contingencies
FR3: Area utilization
FR4: Convenience for counting
FR5: Easiness in retaining FIFO
FR6: Requirement of administration and system support

FR1, FR2, FR3, FR4, and FR5 are benefit attributes whereas FR6 is a cost attribute.

The numerical approximation system for intangible objectives defined as in Table 3.2 and membership functions of the linguistic variables are shown in Fig. 3.7.

The numerical approximation system for tangible objectives is defined as in Table 3.3 and membership functions of linguistic variables are in Fig. 3.8.

Table 3.2 Fuzzy linguistic scal for intangible objectives

Linguistic variable	Triangular fuzzy numbers
Poor	(0, 0, 4)
Fair	(2, 4, 6)
Good	(4, 6, 8)
Very good	(6, 8, 10)
Excellent	(8, 10, 10)

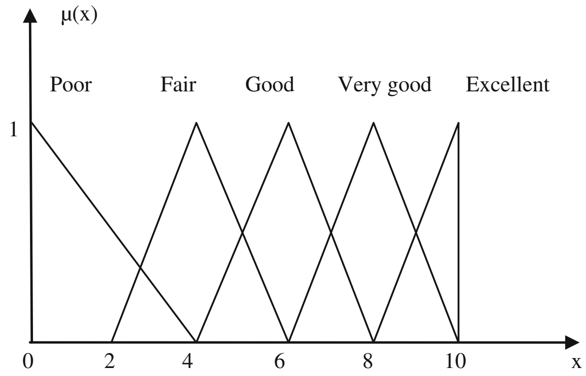


Fig. 3.7 The numerical approximation system for intangible factors

Table 3.3 Fuzzy Linguistic scale for tangible

Linguistic variable	Triangular fuzzy numbers
Very low	(0, 0, 4)
Low	(2, 4, 6)
Average	(4, 6, 8)
High	(6, 8, 10)
Very high	(8, 10, 10)

Figure 3.9 shows the design ranges for benefit and cost attributes:

Experts evaluated the storage policies with respect to the functional requirements. These evaluations produce the system range data for storage policies. Each expert informs his/her own linguistic judgment and these judgments are combined using the corresponding triangular fuzzy numbers. The judgments for each objective are summed and their arithmetic mean is calculated. These means are returned to the most representative linguistic terms and given in Table 3.4.

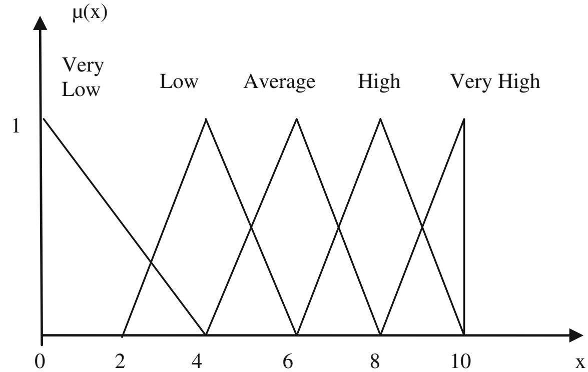
As an example using Figs. 3.7 and 3.9, the information content of time performance in picking load when processing an order of random storage is calculated as follows:

$$\begin{aligned}\text{Common Area} &= (2.5-2)\times 0.25\times 0.5 + (5.5-2.5)\times 0.25 + (6-5.5)\times 0.25\\ &\quad \times 0.5 + (5.5-2.5)\times(0.5-0.25)\times 0.5\\ &= 1.2\bar{5}\end{aligned}$$

$$\text{System area} = (6-2)\times 1/2 = 2$$

$$I = \text{Log}_2\left(\frac{\text{System Area}}{\text{Common Area}}\right) = \text{Log}_2\left(\frac{2}{1.25}\right) = 0.678$$

Fig. 3.8 The numerical approximation system for tangible factors



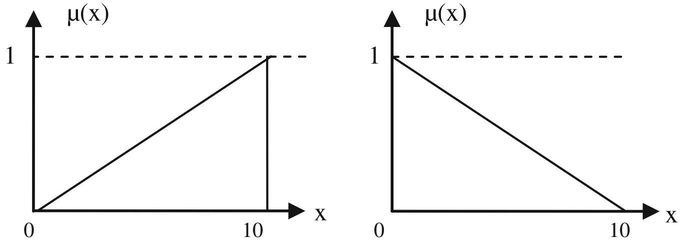


Fig. 3.9 Design ranges for benefit and cost attributes

Table 3.4 The system range data for storage policies

	FR1	FR2	FR3	FR4	FR5	FR6
Random	Fair	Very good	Excellent	Poor	Fair	High
Class-based	Very good	Good	Good	Good	Good	Low
Reserve	Good	Fair	Fair	Fair	Fair	Average
Dedicated	Excellent	Fair	Poor	Excellent	Very good	Very low

Table 3.5 The information content for storage policies

	FR1	FR2	FR3	FR4	FR5	FR6	$\sum I_i$
Random	0.678	0.061	0	1.807	0.678	1.585	4.809
Class-based	0.061	0.263	0.263	0.263	0.263	0.263	1.376
Reserve	0.263	0.678	0.678	0.678	0.678	0.678	3.653
Dedicated	0	0.678	1.807	0	0.061	0	2.546

The information contents for the functional requirements with respect to the storage policy alternatives are given in Table 3.5. Class-based storage policy is selected with minimum information content.

3.7 Conclusion

In this chapter, fuzzy information axiom model is provided to evaluate storage policies. Functional requirements for storage policies have been incorporated into the ranking and selection process According to the uncertainties influencing the evaluation of storage policies, fuzzy numbers are used to model the problem. The advantage of using fuzzy information axiom is the consideration of changing expectations of decision makers for storage policies. The most appropriate storage policy has been identified for a manufacturing firm within the application.

The inspiration for this chapter is not only to introduce background information about storage policies and storage assignment models, but also to present decision support to the planners in warehouses for storage policy selection via a systematic approach. Therefore, this study is thought to contribute to industry in terms of effectively raising decision-making capability of businesses for storage related activities.

As a future research topic, different fuzzy decision-making methods can be compared for storage policy selection problem. Another research direction would be to include weight assignment for functional requirements.

References

Askin RG, Standridge CR (1993) Modeling and analysis of manufacturing systems. Wiley, New York

Bartholdi JJ, Hackman ST (2008) Warehouse and distribution science. The supply chain and logistics institute, Atlanta

Cebi S, Kahraman C (2010) Developing a group decision support system based on fuzzy information axiom. Knowl-Based Syst 23(1):3–16

Chan FTS, Chan HK (2011) Improving the productivity of order picking of a manual-pick and multi-level rack distribution warehouse through the implementation of class-based storage. Expert Syst Appl 38(3):2686–2700

Chen L, Langevin A, Riopel D (2010) The storage location assignment and interleaving problem in an automated storage/retrieval system with shared storage. Int J Prod Res 43(9):1785–1805

De Koster R, Le-Duc T, Roodbergen KJ (2007) Design and control of warehouse order picking: a literature review. Eur J Oper Res 182:481–501

Eddy VM (2004) A simulation study of warehouse storage assignments, Master's Thesis. Worcester Polytechnic Institute, Worcester

Eynan A, Rosenblatt MJ (1994) Establishing zones in single-command class-based rectangular AS/RS. IIE Trans 26(1):38–46

Francis RL, White JA (1974) Facility layout and location-an analytical approach. Prentice-Hall, New Jersey

Frazelle E (2002) World-class warehousing and material handling. McGraw-Hill, New York

Goetschalckx M, Ratliff HD (1990) Shared storage policies based on the duration stay of unit loads. Manage Sci 36(9):1120–1132

Goetschalekx M (2012) Storage systems and policies. In: Riccardo M (ed) Warehousing in the global supply chain: advanced models, tools and applications for storage systems. Springer, London

Graves SC, Hausman WH, Schwarz LB (1977) Storage-retrieval interleaving in automatic warehousing systems. Manage Sci 23(9):935–945

Gu J, Goetschalckx M, McGinnis LF (2007) Research on warehouse operation: a comprehensive review. Eur J Oper Res 177(1):1–21

Gu J, Goetschalckx M, McGinnis LF (2010) Research on warehouse design and performance evaluation: a comprehensive review. Eur J Oper Res 203:539–549

Hartley RVL (1928) Transmission of Information. Bell Syst Tech J 7(3):535–563

Hassini E (2008) Storage space allocation to maximize inter-replenishment times. Comput Oper Res 35:2162–2174

Hausman WH, Schwarz LB, Graves SC (1976) Optimal storage assignment in automatic warehousing systems. Manage Sci 22(6):629–638

Heskett JL (1963) Cube-per-order index—a key to warehouse stock location. Transp Distrib Manage 3:27–31

Kong C, Masel D (2008) Methods for design and management of a fast-pick area in a warehouse. In: Proceedings of the 2008 industrial engineering research conference, Vancouver, British Columbia, CA, pp. 1367–1372

Kovacs A (2011) Optimizing the storage assignment in a warehouse served by milkrun logistics. Int J Prod Econ 133:312–318

Kulturel S, Ozdemirel NE, Sepil C, Bozkurt Z (1999) Experimental investigation of shared storage assignment policies in automated storage retrieval systems. IIE Trans 31:739–749

Li M, Chen X, Liu C (2008) Pareto and niche genetic algorithm for storage location assignment optimization problem. In: Proceedings of the 3rd international conference on innovative computing information and control. ISBN: 978-0-7695-3161-8/08

Lee MK, Elsayed, EA (2005) Optimization of warehouse storage capacity under a dedicated storage policy. Int J Prod Res 43(9):1785–1805

Montulet P, Langevin A, Riopel D (1998) Minimizing the peak load: an alternate objective for dedicated storage policies. Int J Prod Res 36(5):1369–1385

Mulcahy DE (1994) Warehouse distribution and operations handbook. McGraw-Hill Education,New York

Muppant VRM, Adil GK (2008a) A branch-and-bound algorithm for class based storage location assignment. Eur J Oper Res 189:492–507

Muppant VRM, Adil GK (2008b) Class-based storage-location assignment to minimise pick travel distance. Int J Logist Res Appl 11(4):247–265

Muppant VRM, Adil GK (2008c) Efficient formation of storage classes for warehouse storage location assignment: a simulated annealing approach. Omega 36:609–618

Pan JCH, Wu MH (2012) Throughput analysis for order picking system with multiple pickers and aisle congestion considerations. Comput Oper Res 39:1661–1672

Park C, Seo J (2009) Mathematical modeling and solving procedure of the planar storage location assignment problem. Comput Ind Eng 57:1062–1071

Park C, Seo J (2010) Comparing heuristic algorithms of the planar storage location assignment problem. Transp Res Part E 46:171–185

Petersen CG (1997) An evaluation of order picking routing policies. Int J Oper Prod Manage 17(1):1096–1111

Petersen CG, Aase G (2004) A comparison of picking, storage, and routing policies in manual order picking. Int J Prod Econ 92:11–19

Roodbergen KJ, Vis, IFA (2009) A survey of literature on automated storage and retrieval systems. Eur J Oper Res 194(2):343–362

Rosenblatt MJ, Eynan A (1989) Deriving the optimal boundaries for class-based automatic storage/retrieval systems. Manage Sci 35(12):1519–1524

Rouwenhorst B, Reuter B, Stockrahm V, van Houtum GJ, Mantel RJ, Zijm WHM (2000) Warehouse design and control: framework and literatures review. Eur J Oper Res 122(3):515–533

Shannon CE (1948) A Mathematical Theory of Communication. Bell Syst Tech J 27(3):379–423

Suh NP (1990) The Principles of Design. Oxford University Press, New York

Suh NP (1995) Design and operations of large systems. J Manuf Syst 14(3):203–213

Suh NP (2001) Axiomatic design: advances and applications. Oxford University Press, New York

Sule DR (1994) Manufacturing facilities: location, planning and design. PWS Publishing, Boston

van den Berg JP, Sharp GP (1998) Forward-reserve allocation in a warehouse. Eur J Oper Res 111(1):98–113

van den Berg JP (1996) Class-based storage allocation in a single command warehouse with space requirement constraints. Int J Ind Eng 3(1):21–28

van den Berg JP (1999) A literature survey on planning and control of warehousing systems. IIE Trans 31(8):751–762

van den Berg JP, Zijm WHM (1999) Models for warehouse management: classification and examples. Int J Prod Econ 59:519–528

van den Berg JP, Gademann AJRM (2000) Simulation study of an automated storage/retrieval system. Int J Prod Res 38(6):1339–1356

Chapter 4
Multi-objective Optimization Approach to Product-planning in Quality Function Deployment Incorporated with Fuzzy-ANP

S. Mungle, S. Saurav and M. K. Tiwari

Abstract Technological innovations and changing customer trends brought by globalization has led tough competition among various industries throughout the globe. Their assiduous efforts to develop new product is crucial for survival. To overcome this problem and to develop a quality product that generates revenue, a dynamical multi-objective evolutionary algorithm(DMOEA) incorporated with quality function deployment (QFD) and fuzzy analytic network process (FANP) is proposed. The proposed approach considers goals such as new product development (NPD) time and cost, technological advancement, and manufacturability for selection of the most suitable product technical requirements (PTRs). A case study of software development is included to demonstrate the effectiveness of the proposed approach and the obtained results are discussed.

Keywords Multi-objective optimization · Product planning · Quality function deployment · Fuzzy analytic network process

4.1 Introduction

Over the last few decades, industries throughout the world have moved towards a new style of doing business based on overseas competitive compulsion, need for global economics, and advancements in technology. Issues such as performance, aesthetics, delivery, quality, and cost in developing their products need to be considered. The wants (like-to-have), needs (must-have), and desires (wish-to-have)

S. Mungle (✉)
University of Arizona, Tucson, USA
e-mail: santoshmungle@gmail.com

S. Saurav · M. K. Tiwari
Indian Institute of Technology, Kharagpur, India

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_4,

of their customers as completely as possible must be known to them (Ho et al. 1999). Planning becomes essential in designing and manufacturing the products efficiently at competitive cost within a short period of time over those offered by competitors (Chen et al. 2004). Product planning is a process to express customer requirements to define a product's feature.

Quality function deployment (QFD) has a significant role to play in the product planning process. QFD, originated in Japan in the 1960s to support the product design process for designing large ships, is a concept and mechanism for translating the 'Voice of Customer' (VoC) through various stages of planning, engineering, and manufacturing into a final product (Akao 1990). Expeditious response is best achieved by reducing the development time of new products and services. QFD helps to diminish development cycle time by reducing implementation errors, enhancing communication, and supporting concurrent engineering. QFD has successfully evinced the reduction of development time by one-half to one-third (Akao 1990).

However, time-honored QFD has its definite limitations and thus many modifications have been proposed in QFD models. When researchers discovered that QFD alone is incompetent, analytic hierarchy process (AHP) was incorporated to determine the degree of importance of the customer requirements (CRs) (Lu et al. 1994; Park and Kim 1998; Armacost et al. 1994; Fukuda and Matsuura 1993). QFD is a group decision-making process and to generalize the opinions of multiple decision makers is a difficult assignment that needs to be tackled (Lee et al. 2009). To overcome this situation, fuzzy AHP approach was implemented along with QFD (Kwong and Bai 2002). In the last decades, researcher used fuzzy set theory for the appraisal of customer needs (Kim et al. 2000; Shen et al. 2001; Chan et al. 1999). However, AHP has the limitation that the interrelationship among the CRs and PTRs cannot be handled using this method. Therefore, ANP technique is applied, which is a generalization of AHP along with fuzzy set theory (Buyukozkan et al. 2004; Ertay et al. 2005; Kahraman et al. 2006). The motive behind the use of ANP is to look after the interrelationships among the CRs and the PTRs. On the other hand, human decision-making often contains ambiguity and uncertainty (Einhorn and Hogarth 1986). Conventional ANPs are inadequate to explicitly capture the importance assessment of CRs and PTRs. To overcome this limitation, Lee et al. (2009) employed a fuzzy supermatrix approach along with QFD on a case study of product design process of backlight unit (BLU) in a thin film transistor liquid crystal display (TFT-LCT) industry. In addition to the QFD-FANP analysis, they considered other objectives namely NPD time, NPD cost, technological advancement, and manufacturability to select the most suitable PTRs that translate the CRs into the product design. This transforms the problem into a multi-objective optimization problem. To resolve this problem in product planning, previously, the researcher used a goal programming approach such as zero–one goal programming and multi-choice goal programming (Karsak et al. 2002; Lee et al. 2009).

However, goal programming considers the human rating which may be imprecise and erroneous. This may lead the decision makers away from the true solution. To overcome this limitation in decision making, this chapter proposes DMOEA along with fuzzy-ANP (FANP) and QFD to resolve product planning problem efficiently. DMOEA yields the true solution by avoiding the generality and predilection over the objectives. DMOEA uses a principal of minimal free energy in thermodynamics. The main features of DMOEA are: (a) A new fitness assignment strategy by amalgamation of Pareto dominance-relation, Gibbs Entropy, and density estimation (b) A metropolis criterion of simulated annealing algorithm and density estimation for selection of new individuals to maintain the diversity of the population (Zou et al. 2008).

The rest of the chapter is organized as follows. Section 4.2 presents a brief description of house of quality (HOQ). Section 4.3 reviews the ANP and super-matrix approach. Section 4.4 shows steps in evaluating the product planning framework and how we move from QFD to MOEA. Section 4.5 shows an implementation steps of the DMOEA. Section 4.6 includes a case study of software development to demonstrate the effectiveness of the proposed approach. Section 4.7 includes the results and discussions. Finally, Section 4.8 concludes the chapter.

4.2 House of Quality

HOQ is a diagram used for defining the relationship between the CRs and the PTRs (Griffin and Hauser 1993; Hauser and Clausing 1988).The seven elements of the HOQ are shown in Fig. 4.1 and are described as follows:

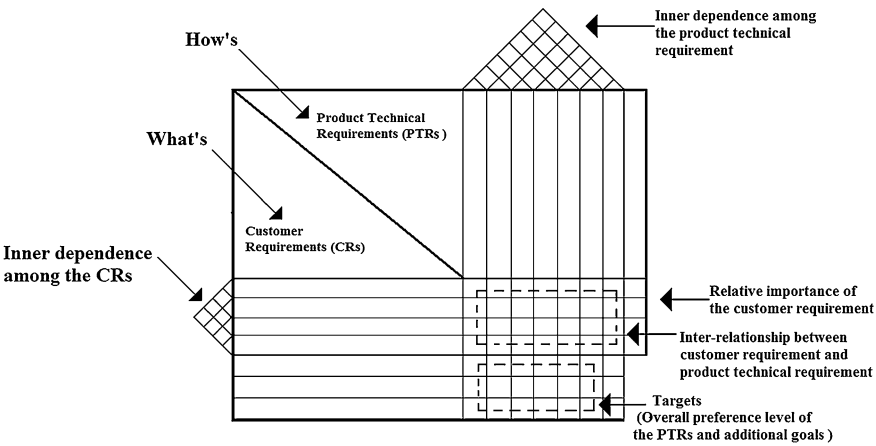


Fig. 4.1 House of quality

1. Customer requirements (Whats?): They are also known as a voice of the customers, customer needs, customer attributes, or demand of quality. The inceptive steps in forming the HOQ involve clarifying and specifying the customers' needs. It is of utter importance to translate the desires of each customer into some tangible values that can be turned into the PTRs. Organizations may use existing data from a market research, or conduct new studies or questionnaire to gather necessary information. In any event, the requirements, which were elucidated and then explicitly stated, should be gratified to the best of that organization's ability.
2. PTRs (Hows?): The next step of the QFD process is identifying what the customer wants and what steps must be initiated to satisfy these wants. PTRs are attributes about the product or service, from an engineering perspective that can be measured and benchmarked against the competition.
3. Interrelationships between CRs and PTRs: The relationship matrix is where the team identifies the relationship between customer needs and the firm's ability to meet those needs, and establishes a correlation between both. The relations can either be presented in a numbers or symbols. In this chapter, symbols have been used to denote the relationship between the whats and hows. The symbols used are shown in Fig. 4.2.
4. Relative importance of the CRs: The collected and organized data from the customer usually consist of too many requisitions to deal with, simultaneously, they must be given weightage or preference. Customers are surveyed for each Whats using 5, 7, or 9 point scales. More detailed 1–10 and anchored scale can also be used. The triangular fuzzy numbers used in this chapter are given in Table 4.1.
5. Inner dependence among CRs: Practically, CRs have inner dependence among themselves or one has an impact on another, positive or negative. These supporting and conflicting requirements can be identified via correlation matrix.
6. Inner dependence among the PTRs: Just like the CRs have inner dependence among themselves, similarly, we have to consider the inner dependence among the PTRs. In order to serve this purpose, we use the ANP technique in this chapter. The HOQ's roof matrix is used to specify the various PTRs that have to be improved collaterally. Symbols as described in Fig. 4.2 are used to compare the dependence of one PTR on another.
7. Targets: It represents the overall preference level of the PTRs and additional goals. The results obtained from the previous steps are used here to estimate the final preference level of the PTRs. Additional design metrics such as a cost, extendibility, manufacturability, etc., can also be incorporated into the analysis

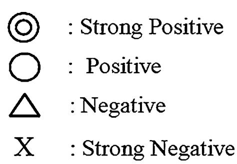


Fig. 4.2 Symbols used in HOQ

Table 4.1 Triangular fuzzy numbers

Linguistic variables	Positive triangular fuzzy numbers	Positive reciprocal triangular fuzzy numbers
Equally important	(1, 1, 1)	(1, 1, 1)
Weakly important	(1, 1, 3)	(1/3, 1, 1)
Moderately important	(1, 3, 5)	(1/5, 1/3, 1)
Important	(3, 5, 7)	(1/7, 1/5, 1/3)
Very important	(5, 7, 9)	(1/9, 1/7, 1/5)
Extremely important	(7, 9, 9)	(1/9, 1/9, 1/7)

at this step (Shillito 1994). These additional design metrics help in attaining a better outcome for deciding the preference level of PTRs and achieve successful improvements in the product that can help a firm in maintaining and sustaining its position in this competitive market.

A structured communication device, the HOQ (Shillito 1994), is constructed using the seven elements mentioned above. With its design-oriented nature, the HOQ serves not only as an expensive resource for designers but also as a way to epitomize and convert feedback from the customers into an information for the engineers. Hence, the HOQ strengthens vertical and horizontal communications. Once having identified crucial PTRs that demand change, they will be driven to the subsequent stage, that is, the many-objective optimization approach along with other design metrics in order to get the final suitable PTRs. Thus finally, the firm has a product, fulfilling both CRs as well as producer requirements within a shorter development time in its hands (Karsak et al. 2002).

4.3 ANP and Supermatrix

4.3.1 ANP

The ANP is a multi-criteria theory of measurement used to derive relative priority scales of absolute numbers from individual judgments that also belong to fundamental scale of absolute numbers (Saaty 1980). Here, one has the liberty to consider any kind of relationship without making assumptions about the independence of higher level elements from lower level elements and about the independence of the elements within a level in the hierarchy, a generalized form of AHP. The AHP is a well-known engineering tool that decomposes a problem into several levels in such a way that they form a hierarchy. The ANP can be used as an effective tool in those cases where the interactions among the elements of a system form a network structure (Saaty 1996).

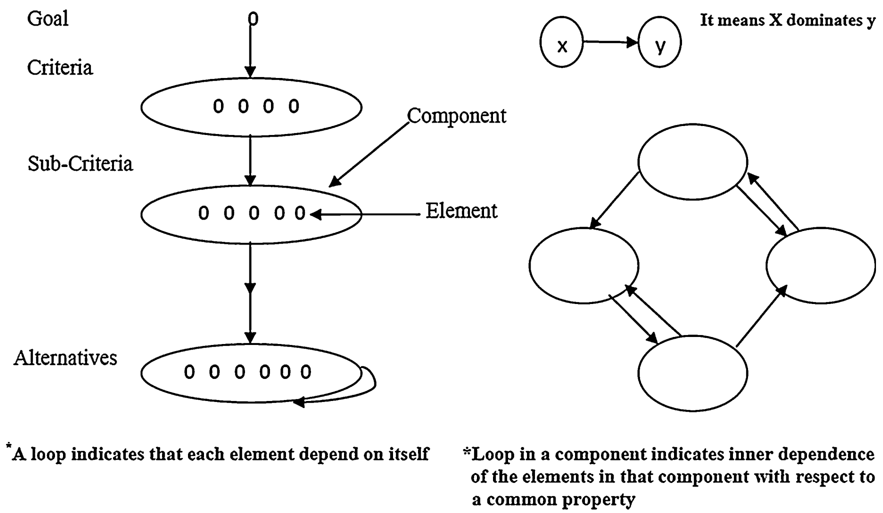


Fig. 4.3 A linear hierarchy and a nonlinear network

ANP gives weightage to interrelationship among the decision levels in a more general form and that makes it worth over AHP which only considers unidirectional hierarchical relationship among the decision levels. The difference in AHP and ANP can be better figured out by the schematic diagram in Fig. 4.3. A hierarchy has a goal or a source node and a sink node at the bottom. It is a lineal composition from top to bottom with no response from lower levels to higher levels but has a loop at the bottom level which signifies that each alternative in that level depends on itself and the elements are independent of each other. Unlike a hierarchy, a network propagates in all directions and its clusters of elements are not organized in a specific order.

4.3.2 Supermatrix

The main three kinds of components are (i) Source component, which is not dominated by any other component, (ii) Sink component which does not dominate others, and (iii) the intermediate components known as transient component that lies in the midst of the above-mentioned categories. Figure 4.4 represents the types of components in a structure. All the interactions among the elements should be considered while structuring out the problem. The supermatrix of system of *N* clusters is denoted as follows:

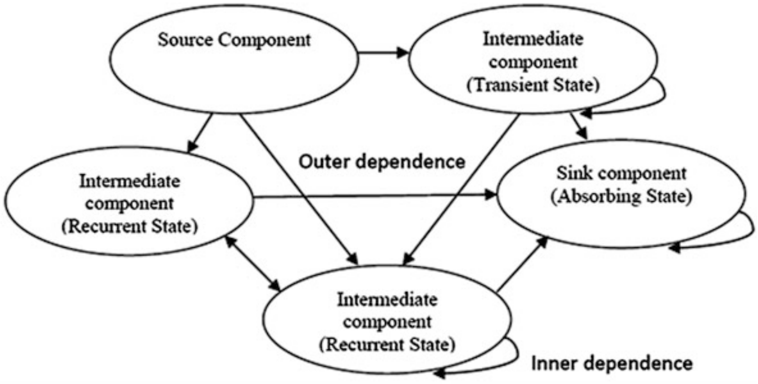


Fig. 4.4 Types of components in a structure

$$W = \begin{array}{cc} & \\ C_1 & \begin{array}{c} e_{11} \\ \vdots \\ e_{1n_1} \end{array} \\ \vdots & \vdots \\ C_k & \begin{array}{c} e_{k1} \\ \vdots \\ e_{kn_k} \end{array} \\ \vdots & \vdots \\ C_N & \begin{array}{c} e_{N1} \\ \vdots \\ e_{Nn_N} \end{array} \end{array} \begin{array}{c} \begin{array}{ccccc} C_1 & \cdots & C_k & \cdots & C_N \\ e_{11}\ldots e_{1n_1} & \cdots & e_{kn_1}\ldots e_{kn_k} & \cdots & e_{N1}\ldots e_{Nn_N} \end{array} \\ \begin{bmatrix} W_{11} & \cdots & W_{1k} & \cdots & W_{1N} \\ \vdots & & \vdots & \ddots & \vdots \\ W_{k1} & \cdots & W_{kk} & \cdots & W_{kN} \\ \vdots & & \vdots & \ddots & \vdots \\ W & \cdots & W & \cdots & W_{NN} \end{bmatrix} \end{array} \tag{4.1}$$

where C_k is the kth cluster, which has n_k element denoted as e_{k1}, $e_{k2}, \ldots, e_{knk}$ whereas the supermatrix of hierarchy with three levels is represented as follows:

$$z = \begin{array}{c} \mathrm{G} \\ \mathrm{CR} \\ \mathrm{PTR} \end{array} \begin{array}{c} \begin{array}{ccc} \mathrm{G} & \mathrm{CR} & \mathrm{PTR} \end{array} \\ \begin{bmatrix} I & & \\ W_{21} & W_{22} & \\ & W_{32} & W_{33} \end{bmatrix} \end{array} \tag{4.2}$$

where W_{21} vector represents impact of the goal on CRs, W_{22} indicates the interdependency of CRs, W_{32} represents the impact of CRs on PTRs, W_{33} represents interdependency of PTRs, and I is the identity matrix and entries with zero signify that the elements have no authority (Saaty 1996). This is also known as unweighted supermatrix.

The unweighted supermatrix must be transformed first to be stochastic, that is, each column of the matrix sums to unity (Saaty 1996). A recommended approach is to determine the relative importance of the clusters in the supermatrix with the column cluster (block) as the controlling component (Saaty 1996). Another common approach is to give equal weights to the blocks in the same column and to make each column sum to unity (Lee et al. 2008). The resulted supermatrix is known as the weighted supermatrix, which is stochastic. The weighted supermatrix is raised to the power of $2\rho + 1$ to achieve a convergence on the importance weights, where ρ is an arbitrarily large number (Saaty 1996). This new matrix is called the limit supermatrix.

4.4 From QFD to MOEA

A people accountable for product planning can belong to different sections within the firm. This is schemed to ensure that many different perspectives are incorporated from the beginning. Later on multi-objective optimization approach is brought into the picture to consider other design metrics to finalize the PTRs. It starts with construction of HOQ as mentioned in Fig. 4.1 and all the CRs and PTRs are mentioned in their respective places and interrelationships between them are evaluated through a brainstorming session conducted by the decision makers appointed by the firms or other method applied by them. Based on the constructed HOQ, the QFD network is constructed as shown in Fig. 4.5.

From the relationship among elements determined from the previous stage and QFD network constructed, a questionnaire is prepared, and all the relationships among the elements must be included. Pairwise comparison among the elements using the linguistic terms of Table 4.1 are done by the decision team. These linguistic variables obtained via pairwise comparison of each part of the

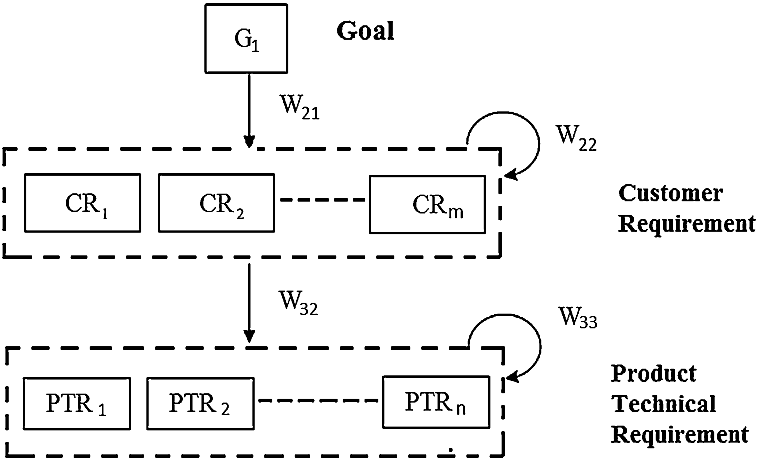


Fig. 4.5 QFD network

questionnaire from each decision maker are transmuted into triangular fuzzy numbers. For example, the following matrix (A_k) can be acquired for decision maker k with pairwise comparison of CRs with respect to the overall objective.

$$A_k = \begin{matrix} CR_1 \\ CR_2 \\ \vdots \\ CR_m \end{matrix} \overset{\begin{matrix} CR_1 & CR_2 & \dots & CR_m \end{matrix}}{\begin{bmatrix} 1 & a_{12k} & \dots & a_{1mk} \\ 1/a_{12k} & 1 & \dots & a_{2mk} \\ \vdots & \vdots & \ddots & \vdots \\ 1/a_{1mk} & 1/a_{2mk} & \dots & 1 \end{bmatrix}} \tag{4.3}$$

where, m represents the number of CRs. Since we cannot take our decision based on one decision maker, a group of people is involved in the decision making to get an aggregated view. Suppose there is a k decision maker in the panel appointed by the firm, a total of k pairwise comparison matrixes are formed and for each pairwise comparison between two elements, there are k triangular fuzzy numbers. In order to get an aggregated result, geometric mean approach is employed to obtain synthetic triangular fuzzy number.

$$\tilde{a}_{ij} = \left(\tilde{a}_{ij1} * \tilde{a}_{ij2} * \tilde{a}_{ij3} * \dots * \tilde{a}_{ijk}\right)^{1/k} \tag{4.4}$$

where,

$$\tilde{a}_{ijk} = \left(x_{ijk}, y_{ijk}, z_{ijk}\right)$$

Hence, fuzzy aggregated pairwise comparison matrix is generated using the above formula.

$$\tilde{A} = \begin{bmatrix} 1 & \tilde{a}_{12} & \dots & \dots & \dots & \dots & \tilde{a}_{1i} \\ 1/\tilde{a}_{12} & 1 & \dots & \dots & \dots & \dots & \tilde{a}_{2j} \\ \vdots & \vdots & 1 & \dots & \dots & \dots & \dots \\ \vdots & \vdots & \vdots & 1 & \tilde{a}_{ij} & \dots & \dots \\ \vdots & \vdots & \vdots & 1/\tilde{a}_{ij} & 1 & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & 1 & \dots \\ 1/\tilde{a}_{1i} & 1/\tilde{a}_{2j} & \dots & \dots & \dots & \dots & 1 \end{bmatrix} \tag{4.5}$$

where $\tilde{a}_{ij} = \left(x_{ij}, y_{ij}, z_{ij}\right)$.

The subsequent step is to defuzzify this matrix. Several methods have been proposed in the literature to defuzzify a fuzzified aggregated pairwise comparison matrix. In this chapter, Center of Gravity method is used to defuzzify the comparison between element i and j (Yager 1978; Klir and Yuan 1995).

$$a_{ij} = \left[\left(x_{ij} + y_{ij} + z_{ij}\right)/3\right] \tag{4.6}$$

Therefore, the defuzzified aggregated pairwise comparison matrix formulated is:

$$A = \begin{bmatrix} 1 & a_{12} & \ldots & \ldots & \ldots & \ldots & a_{1i} \\ 1/a_{12} & 1 & \ldots & \ldots & \ldots & \ldots & a_{2j} \\ \vdots & \vdots & 1 & \ldots & \ldots & \ldots & \ldots \\ \vdots & \vdots & \vdots & 1 & a_{ij} & \ldots & \ldots \\ \vdots & \vdots & \vdots & 1/a_{ij} & 1 & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots & \ldots & 1 & \ldots \\ 1/a_{1i} & 1/a_{2j} & \ldots & \ldots & \ldots & \ldots & 1 \end{bmatrix} \tag{4.7}$$

The next step is to compute a local priority vector for each defuzzified aggregated pairwise comparison matrix as an estimate of relative importance associated with the elements being compared by solving the following equation (Saaty 1980, 1996).

$$A.w = \lambda_{\max}.w \tag{4.8}$$

where A represents defuzzified aggregated pairwise comparison matrix, w represents an eigenvector, and $\lambda_{\max}$ signifies the largest eigenvalue of A. Forming the defuzzified aggregated pairwise comparison matrix is not enough as its consistency property has to be checked by examining consistency index (CI) and consistency ratio (CR) (Saaty 1980).

$$CI = (\lambda_{\max} - n) / (n - 1) \tag{4.9}$$

$$\mathrm{CR} = \mathrm{CI}/\mathrm{RI} \tag{4.10}$$

where n is the number of items being compared in the matrix and RI is the random index, the average CI of a randomly generated pairwise comparison matrix of similar size (Saaty 1980).

The matrix is considered to be inconsistent if a CR value exceeds the threshold value. In such a case, decision makers need to recheck the original values in the specific part of the questionnaire. Then all the priority vectors are placed at the appropriate places to form unweighted supermatrix which is then converted into weighted supermatrix and made stochastic and finally into limit supermatrix as discussed above. Then the final priority list of the PTRs is obtained whose value is used to formulate the objective function for DMOEA.

4.5 Fundamentals of Multi-objective Optimization

This section briefly summarizes the fundamental concepts of multi-objective optimization. The general multi-objective optimization problem, in a minimization case, can be formulated as follows:

$$\begin{aligned} &\text{Min}\, f(x) = \{f_1(x), f_2(x), \ldots, f_m(x)\} \\ &\text{s.t} \qquad\qquad\qquad\qquad x \in D, \end{aligned} \tag{4.11}$$

where $m(m \geq 2)$ is the number of objectives, $x = (x_1, x_2, \ldots, x_n)$ is the vector representing the decision variables, and D is the set of feasible solutions.

A multi-objective optimization problem usually has not one unique optimal solution, but a set of solutions known as the Pareto-optimal set. Each Pareto-optimal solution represents a compromise between different objectives, and the component of the corresponding vector of objectives cannot be all simultaneously improved. Two concepts are indeed of great importance in multi-objective optimization: Pareto dominance and Pareto optimality are defined as follows.

Definition 4.1 (*Pareto dominance*) A given vector $u \in D$ dominates a vector $v \in D$in the Pareto sense, if and only if u is partially less than$v(u \prec v)$, i.e.,

$$\begin{cases} f_i(u) \leq f_i(v) & \text{for all} \quad i \in \{1, \ldots, m\} \\ f_i(u) < f_i(v) & \text{for at least one} \quad j \in \{1, \ldots, m\} \end{cases}$$

Definition 4.2 (*Pareto-optimal solution*) A solution $u \in D$ is a Pareto optimal solution, if and only if there is no $v \in D$ such that v dominates u. Pareto-optimal solutions are also called non-dominated solutions.

Definition 4.3 (*Pareto-optimal set*) The Pareto-optimal set is defined as

$$P = \{x \in D : x \text{ is a Pareto-optimal solution in } D\}$$

Definition 4.4 (*Pareto front*) The Pareto front is defined as $PF = \{f(x) : x \in P\}$

where P is the Pareto-optimal set.

4.5.1 DMOEA for Multi-objective Optimization

The main focus of this chapter is to propose multi-objective optimization technique to determine the suitable PTRs to be considered in the design process. In recent years, such types of problems are tackled with Multi-choice goal programming in which goals are set and priorities to each goal are arbitrarily incurred to optimize total weighted deviation from all goals (Lee et al. 2009). Considering appropriate weight for each goal which reflects true weightage to the overall deviation weight is still an unresolved problem for the decision maker. From previous description, we can assert that the multi-choice goal programming is not efficient to solve such kind of complex problems due to it's incapability of reflecting appropriate weight pertaining to each goal.

In the last decade, many metaheuristics were proposed by the researcher to solve the multi-objective optimization problem with two or three objective functions. Metaheuristics like non-dominated sorting genetic algorithm (NSGA-II) (Deb et al. 2002), multi-objective genetic algorithm (MOGA) (Fonseca and Fleming 1993),, strength Pareto evolutionary algorithm-2 (SPEA-2) (Zitzler et al. 2001), and Pareto evolutionary archiving strategy (PAES) (Knowles and Corne 1999) have been widely used in the academic and industrial fields. However, these metaheuristics have been shown to be unsuited to solve optimization problems with more than three objectives. A large number of objectives pose new challenges to algorithm design, visualization, and implementation (Fleming et al. 2005). In recent years, almost all hot issues in the design of MOEAs have been related to the handling of a large number of objectives (Koppen and Yoshida 2007; Fleming and Purshouse 2007). In this chapter, we adapted the DMOEA approach developed by Zou et al. (2008). The DMOEA introduces a two new features namely fitness allocation scheme and selection criteria. The following subsections detail the fitness allocation scheme and selection criterion used in the DMOEA.

4.5.1.1 Fitness Allocation Scheme

Fitness is adopted as an aggregating function that combines ranking with gibbs entropy and density. The fitness allocation scheme is analogous to gibbs free energy or gibbs function. In thermodynamics, the gibbs free energy is thermodynamic potential that measures useful or process-initiating work obtainable from an isothermal, isobaric thermodynamic system. The gibbs free energy is defined as

$$G = <H> - TS \tag{4.12}$$

where H is enthalpy, S is entropy, and T is temperature. Minimization of G is proportional to the minimization of H and maximization of TS. It is widely known as the "Principle of minimal free energy."

Gibbs free energy equation has been predominantly used in various fields. The minimization of the objective function (convergence toward the Pareto-optimal set) and density estimation are included in new fitness allocation scheme as selection criteria when two members of the population belong to the same Pareto rank. In order to encapsulate a multi-objective optimization problem into a Gibbs statistical framework, we combine the rank value $R(i)$ calculated by Pareto-dominance relation with Gibbs entropy $S(i)$ to assign a new fitness scheme $F(i)$ for each individual i in the population, that is,

$$F(i) = R(i) - TS(i) - d(i) \tag{4.13}$$

where $R(i)$ is the rank of individual i which is equal to the number of solutions n_i that dominate solution i. In this way, $R(i) = 0$ is corresponding to a non-nominated individual, while high $R(i)$ value means that i is dominated by many individuals, and $d(i)$ is the crowding distance which is computed by using the density estimation technique described in Deb et al. (2002).

$$S(i) = -p_T(i) \log p_T(i) \tag{4.14}$$

where $p_T(i) = \frac{1}{z} \exp\left(-\frac{R(i)}{T}\right)$ is analogous to the Gibbs distribution.

$z = \sum_{i=1}^{N} \exp\left(-\frac{R(i)}{T}\right)$ is called the partition function and N is the population size.

4.5.1.2 Selection Criterion

Each individual in initial population, $P(t)$ is assigned a fitness value as per Equation numbers (4.15–4.19), where t denotes generation index and then it is sorted in an increasing order. The individual with high fitness value is considered to be "worse individuals" and that with low fitness value is considered to be "best individuals" considering the minimization of the objectives. In every generation new individuals are generated by genetic operators. The worst individuals $W(t)$ formed are compared with the new individuals $R(t)$ obtained after the genetic operations, keeping in mind that each individual is compared once. Since DMOEA is based on the thermodynamic principle, we seek to exploit the Metropolis criterion of simulated annealing algorithm and the estimation density to guide the select process, that is,

1. If $R(X_{\text{new}}) < R(X_{\text{worse}})$, then $X_{\text{worst}} = X_{\text{new}}$
2. If $R(X_{\text{new}}) = R(X_{\text{worse}})$ and $cd(X_{\text{new}}) > cd(X_{\text{worse}})$, then $X_{\text{worst}} = X_{\text{new}}$

where R (X_{new}) is the rank of new individual selected for comparison. R (X_{worse}) is the rank of the worst individual selected randomly from $W(t)$.

4.5.2 DMOEA Steps

1. Randomly generate the initial population $P(0)$ and set generation index $t = 0$.
2. Determine the Pareto rank values $\{R_1(t)\ldots R_N(t)\}$ of all individuals in $P(t)$ as mentioned above, where N is the population size.
3. Evaluate the fitness of each individual in $P(t)$ according to Eqs. (4.15) to (4.19) and sort them in an increasing order.
4. Apply genetic operator to generate new individuals $C(t)$, randomly select $m1$ elements to do multi-parent crossover (Eiben et al. 1994) and $m2$ individuals for mutation to form 'n' child.
5. Record 'n' individuals with highest fitness value as worst individuals $W(t)$.
6. Combine initial population and child. $E(t) = \{P(t), C(t)\}$.
7. Determine the rank of all elements in $E(t)$.

8. Compare the individual of $W(t)$ with the member of child population $C(t)$ as per selection criterion to form new population $P(t + 1)$ of size N.
9. Determine the rank of individuals in $P(t + 1)$ and record the individuals with rank = 0.
10. Repeat the steps from 2 to 9 for next generation ($t = t + 1$) until the stopping criteria is satisfied, i.e., a fixed number of generations, or when no significant improvement in the solution occurs.

4.6 A Case Study of Software Development

In this research, we include the case study of software development to demonstrate the effectiveness of the proposed approach. Through the literature review and interview with experts, we gathered all the possible CRs and PTRs. We identified the most important CRs and PTRs and it is given in Tables 4.2 and 4.3.

The preference level of PTRs calculated by the intermediate steps is discussed in Sect. 4.4 and is shown in Table 4.4. PTR_6 has the highest preference level with priority of 0.1235, followed by PTR_{11} with priority of 0.1111, then PTR_4 with priority of 0.0988, PTR_{17} with priority of 0.0864, and then PTR_8, PTR_5, PTR_{12}, PTR_{15}, PTR_{13}, PTR_9, PTR_3, PTR_{18}, PTR_{14}, PTR_7, PTR_{16}, PTR_{10}, PTR_2, PTR_1 subsequently. The additional goals that are given importance are NPD Cost, Manufacturability, NPD Time, and Technological Advances. Henceforth, many-objective approaches are incorporated after formulation of objective functions.

The cost incurred in developing specific PTRs is known as NPD Cost. Manufacturability is the characteristics considered in the design cycle that focus on

Table 4.2 Customer requirements

Serial number	Customer requirements (CRs)
1	It must be simple to use
2	Easy to install
3	Easy to learn initially
4	Small disk space requirements
5	Automatically saves our data
6	No cost for bug fixing
7	The system must be adaptable
8	Easy to upgrade capabilities
9	Allows us to restart where we stopped
10	Operates with our network
11	Good price/performance
12	Looks good
13	Allows user to write and save useful macros
14	Efficiently uses memory
15	Supports the printer I need

Table 4.3 Product technical requirements (PTRs)

Serial number	Product technical requirements (PTRs)	Units
1	New roles can be added without changing in source code	Number of changes
2	Speed of user interface to fetch data from other systems	Time
3	Layout designed according to standard of customer	Number of deviations
4	Ease of submitting	Minutes
5	Flexibility to store different kinds of code	Number of types of codes that cannot be stored
6	Efficient user interface	Seconds for page to
7	Flexible architecture according to standard of customer	Number of deviations
8	Flexible interface to other systems	Number of chances to interface when new code is added
9	Layered architecture	Changes to code if database is changed to another
10	Support to IE 9.0 and newer versions	Number of differences between versions
11	Interaction time for common operations	Time
12	Estimated execution time to complete common operations	Time
13	Number of deviations from user interface guidelines	Numbers
14	Training time	Time
15	Online preliminary help	Percent
16	Estimated time for novice to install	Time
17	Estimated time for novice to learn	Time
18	System hardware requirements	Number of hardware components

process capabilities, machine or facility flexibility, and the overall ability to consistently produce at the required quality level instead of product elegance. A technological benefit that is obtained from developing specific PTRs is called Technological Advances.

The main focus of this chapter is multi-objective optimization approach, minimizing NPD Cost and NPD Time as well as maximizing results of QFD-FANP, Manufacturability, and Technological Advances. The multi-objective optimization models maybe formulated as:

Maximize QFD-FANP:

$$\begin{aligned} f_2(x) = {} & 0.1081X_1 + 0.0405X_2 + 0.0676X_3 + 0.0135X_4 + 0.0135X_5 + 0.0270X_6 \\ & + 0.0405X_7 + 0.0135X_8 + 0.0541X_9 + 0.0135X_{10} + 0.0270X_{11} + 0.0135X_{12} \\ & + 0.1351X_{13} + 0.0541X_{14} + 0.0811X_{15} + 0.0541X_{16} + 0.1351X_{17} + 0.1081X_{18} \end{aligned} \tag{4.15}$$

Table 4.4 Problem data set

Goal	QFD-FANP	NPD cost	Manufacturability	NPD time	Technological advances
PTR_1	0.0123	0.1081	0.0865	0.0857	0.0875
PTR_2	0.0123	0.0405	0.0577	0.0286	0.0875
PTR_3	0.0494	0.0676	0.0673	0.0476	0.0125
PTR_4	0.0988	0.0135	0.0385	0.0857	0.0875
PTR_5	0.0864	0.0135	0.0288	0.019	0.075
PTR_6	0.1235	0.027	0.0192	0.0762	0.1125
PTR_7	0.0247	0.0405	0.0192	0.0857	0.0875
PTR_8	0.0864	0.0135	0.0962	0.0571	0.0375
PTR_9	0.0494	0.0541	0.0096	0.0571	0.025
PTR_{10}	0.0123	0.0135	0.0865	0.0857	0.0125
PTR_{11}	0.1111	0.027	0.0577	0.0952	0.0875
PTR_{12}	0.0741	0.0135	0.0865	0.0762	0.0875
PTR_{13}	0.0494	0.1351	0.0481	0.0381	0.0375
PTR_{14}	0.0247	0.0541	0.0385	0.0571	0.0625
PTR_{15}	0.0494	0.0811	0.0769	0.0286	0.05
PTR_{16}	0.0123	0.0541	0.0385	0.0571	0.0125
PTR_{17}	0.0864	0.1351	0.0577	0.0095	0.0125
PTR_{18}	0.037	0.1081	0.0865	0.0095	0.025

Minimize NPD Cost:

$$\begin{aligned} f_1(x) = {} & 0.0123X_1 + 0.0123X_2 + 0.0494X_3 + 0.0988X_4 + 0.0864X_5 + 0.1235X_6 \\ & + 0.0247X_7 + 0.0864X_8 + 0.0494X_9 + 0.0123X_{10} + 0.1111X_{11} + 0.0741X_{12} \\ & + 0.0494X_{13} + 0.0247X_{14} + 0.0494X_{15} + 0.0123X_{16} + 0.0864X_{17} + 0.0370X_{18} \end{aligned} \tag{4.16}$$

Maximize Manufacturability:

$$\begin{aligned} f_3(x) = {} & 0.0865X_1 + 0.0577X_2 + 0.673X_3 + 0.0385X_4 + 0.0288X_5 + 0.0192X_6 \\ & + 0.0192X_7 + 0.0962X_8 + 0.0096X_9 + 0.0865X_{10} + 0.0577X_{11} + 0.0865X_{12} \\ & + 0.0481X_{13} + 0.0385X_{14} + 0.0769X_{15} + 0.0385X_{16} + 0.0577X_{17} + 0.865X_{18} \end{aligned} \tag{4.17}$$

Minimize NPD Time:

$$\begin{aligned} f_4(x) = {} & 0.0857X_1 + 0.0286X_2 + 0.0476X_3 + 0.0857X_4 + 0.0190X_5 + 0.0762X_6 \\ & + 0.0857X_7 + 0.0571X_8 + 0.0571X_9 + 0.0857X_{10} + 0.0952X_{11} + 0.0762X_{12} \\ & + 0.0381X_{13} + 0.0571X_{14} + 0.0286X_{15} + 0.0571X_{16} + 0.0095X_{17} + 0.0095X_{18} \end{aligned} \tag{4.18}$$

Maximize Technological Advances:

$$f_5(x) = 0.0875X_1 + 0.0875X_2 + 0.0125X_3 + 0.0875X_4 + 0.0750X_5 + 0.1125X_6 + 0.0875X_7 + 0.0375X_8 + 0.0250X_9 + 0.0125X_{10} + 0.0875X_{11} + 0.0875X_{12} + 0.0375X_{13} + 0.0625X_{14} + 0.0500X_{15} + 0.0125X_{16} + 0.0125X_{17} + 0.0250X_{18} \quad (4.19)$$

4.7 Results and Discussions

The main objective of this chapter is proposing an algorithm to select the appropriate or suitable PTRs. In problem formulation, there are 18 PTRs of binary nature leading various combinations of PTRs, where each combination represents five different objective functions, namely maximize QFD-FANP, manufacturability, technological advances, and minimize NPD cost and time. The computational complexity henceforth intensifies due to the conflicting nature among the objectives. To arrive at an exact combination of PTRs which simultaneously satisfies all the objectives with variant optimizing nature, becomes a meticulous task. To resolve this problem, DMOEA has been brought into the scenario. The DMOEA is capable to reduce the large search space by finding non-dominated solution over successive generations. The DMOEA result is sensitive to algorithm parameter and hence, it is required to perform repeated simulations to find a suitable value for the parameters.

In DMOEA, temperature and number of parent for multi-parent crossover is highly sensitive parameter and to find appropriate value of these, several simulation tests were carried out at fixed number of generation and population size. The best parameter for DMOEA selected through several simulation tests is shown in Table 4.5. The best Pareto set obtained out of several simulations run is shown in Fig. 4.6. Figure 4.6 shows the trade-off among five objectives, namely QFD-FANP, manufacturability, technological advances, and NPD cost and time in parallel coordinate plots. Every objective is covered and the structure of the set is almost symmetric, indicating a uniformly spread distribution of solutions over the whole Pareto front. Table 4.6 shows the optimal PTRs selection plan.

The best non-dominated solutions are found at 300 generations of population size 200, as given in Table 4.7. Each of these solutions represents trade-offs among the objectives, namely QFD-FANP, manufacturability, technological advances, and NPD cost and time. This means that one objective cannot be improved without sacrificing another objective. For practical application, the decision maker is interested in selecting one solution that satisfies all the objectives to some extent. Such solution is known as a best compromise solution and is determined by using fuzzy set theory. The following subsection briefly presents the fuzzy set theory.

Table 4.5 Best DMOEA parameter for selection of PTRs in product planning

DMOEA parameter	Parameter value
Population size	200
Number of generations	300
Temperature	35
Number of parent in multi-parent crossover	20
Probability of crossover(P_c)	0.8
Probability of mutation (P_m)	0.1

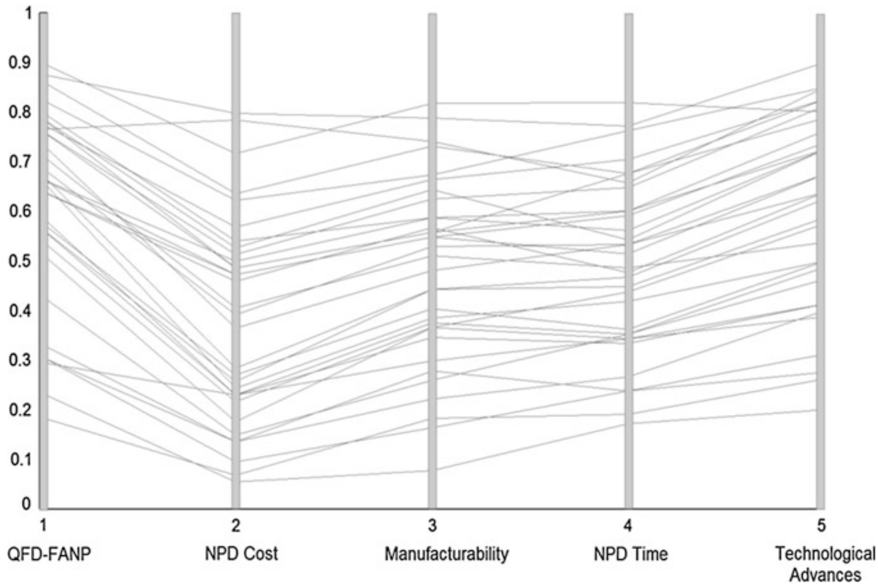


Fig. 4.6 Parallel coordinate plot

4.7.1 Best Compromise Solution

Upon having the non-dominated set, the proposed approach presents a fuzzy-based mechanism to extract a solution as the best compromise. Due to the imprecise nature of the decision maker's judgment, the *i*th objective function of a solution in the Pareto-optimal set is represented by a membership function defined by Sakawa et al. (1987).

$$U_i = \left\{ \begin{array}{ll} 1, & F_i \leq F_i^{\min} \\ \frac{F_i^{\max} - F_i}{F_i^{\max} - F_i^{\min}}, & F_i^{\min} < F_i < F_i^{\max} \\ 0, & F_i \geq F_i^{\max} \end{array} \right\} \tag{4.20}$$

Table 4.6 Optimal PTRs selection plan

Serial number	PTRs selection plan $(X_1, X_2, \ldots, X_{18})$																	
1	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
2	0	0	1	0	1	1	0	0	1	0	1	1	0	0	1	0	1	1
3	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
4	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0
5	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	1	0	1	0	0	0	1	0	1	0	0	0	1	0	1	0
8	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
9	0	0	1	1	1	1	0	0	1	1	1	1	0	0	1	1	1	1
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
12	0	1	0	0	1	1	0	1	0	0	1	1	0	1	0	0	1	1
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	1	1	1	0	0	0	1	1	1	0	0	0	1	1	1	0
15	0	0	0	0	1	1	0	0	0	0	1	1	0	0	0	0	1	1
16	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
17	0	0	1	0	1	0	0	0	1	0	1	0	0	0	1	0	1	0
18	0	0	1	0	1	0	0	0	1	0	1	0	0	0	1	0	1	0
19	1	1	0	0	1	1	1	1	0	0	1	1	1	1	0	0	1	1
20	0	0	1	0	1	0	0	0	1	0	1	0	0	0	1	0	1	0
21	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
24	0	0	1	0	1	0	0	0	1	0	1	0	0	0	1	0	1	0
25	0	0	1	1	1	0	0	0	1	1	1	0	0	0	1	1	1	0
26	0	0	0	1	1	1	0	0	0	1	1	1	0	0	0	1	1	1
27	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0
30	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1	1
31	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	1	0
32	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1
33	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1

where $F_i^{\max}$ and $F_i^{\min}$ are the maximum and minimum values of the ith objective function, respectively. Subsequently, each non-dominated solution k, the normalized membership function U^k is calculated as

$$U^k = \frac{\sum_{i=1}^{N_{\text{obj}}} U_i^k}{\sum_{j=1}^{M} \sum_{i=1}^{N_{\text{obj}}} U_i^j} \tag{4.21}$$

where M is the number of non-dominated solutions. The best compromise solution is the one having maximum value of U_k. As a matter of fact, arranging all solutions in non-dominated set in descending order according to their membership function

Table 4.7 Optimal PTRs selection plan representing fitness value

Serial number	Fitness value				
	QFD-FANP	NPD cost	Manufacturability	NPD time	Technological advances
1	0.5802	0.2701	0.4038	0.3618	0.5
2	0.9012	0.7161	0.8171	0.8188	0.8
3	0.6913	0.3917	0.5288	0.5332	0.675
4	0.6419	0.4728	0.548	0.5903	0.7375
5	0.7654	0.4863	0.5576	0.676	0.7875
6	0.3086	0.1351	0.2788	0.238	0.3125
7	0.6666	0.4998	0.5864	0.5618	0.725
8	0.679	0.2836	0.4423	0.4475	0.5875
9	0.1852	0.0675	0.1827	0.1905	0.2625
10	0.2962	0.2296	0.3749	0.3523	0.4625
11	0.8025	0.5403	0.5865	0.5999	0.7625
12	0.6666	0.4728	0.548	0.5141	0.675
13	0.5185	0.1755	0.3653	0.438	0.575
14	0.7777	0.5674	0.6634	0.7046	0.825
15	0.7901	0.4863	0.5576	0.5998	0.725
16	0.5432	0.2161	0.3654	0.3427	0.3875
17	0.6419	0.4593	0.5672	0.4761	0.6375
18	0.7408	0.3647	0.4807	0.5332	0.6375
19	0.8765	0.7971	0.7884	0.7712	0.9
20	0.7654	0.5133	0.6249	0.6475	0.8125
21	0.6667	0.2431	0.3846	0.4189	0.5
22	0.3333	0.135	0.2211	0.2666	0.4
23	0.2346	0.054	0.0769	0.1714	0.2
24	0.3087	0.0946	0.1634	0.238	0.275
25	0.8271	0.6215	0.673	0.7617	0.85
26	0.716	0.4053	0.5096	0.4855	0.5375
27	0.5926	0.2296	0.2981	0.3427	0.4125
28	0.4321	0.1485	0.2596	0.3523	0.4875
29	0.5679	0.2566	0.4422	0.4666	0.625
30	0.7654	0.7836	0.7403	0.657	0.85
31	0.5679	0.2296	0.3461	0.3332	0.4125
32	0.7901	0.5269	0.6442	0.5427	0.725
33	0.8642	0.635	0.7307	0.676	0.825

will provide the decision maker with a priority list of non-dominated solutions. This will guide the decision maker in view of the current operating conditions. The best compromise solution with PTRs selection plan is given in Table 4.8 and the corresponding objective values are given in Table 4.9. This PTRs selection plan helps the decision maker to select the precise PTRs for modification in the respective product.

In Table 4.9, PTR selection plan represents PTR2, PTR3, PTR4, PTR5, PTR6, PTR7, PTR8, PTR11, PTR12, PTR14, PTR15, PTR17, and PTR18 is selected for

Table 4.8 Best compromise solution

PTRs Selection plan $(X_1, X_2, \ldots, X_{18})$																	
0	1	1	1	1	1	1	1	0	0	1	1	0	1	1	0	1	1

Table 4.9 Best compromise solution representing objective value

QFD-FANP	NPD cost	Manufacturability	NPD time	Technological advances
0.8642	0.635	0.7307	0.676	0.825

implementation, whereas PTR1, PTR9, PTR10, PTR13, and PTR16 are not selected for further modification in product.

4.8 Conclusion

Increase in global competition entails industries to necessitate improvement in the products to satisfy customer demands with generating revenues. In order to fulfill this stipulated condition, product planning becomes essential. This chapter presents a step approach for product planning, comprising QFD-FANP analysis, to provide preference level among PTRs along with additional goals such as NPD cost, NPD time, manufacturability, and technological advances. This makes decision making difficult due to the conflicting nature of additional goals. To confront this tedious task multi-objective optimization technique is brought into the picture to evaluate and select the most appropriate PTRs for product planning. In the past, a few attempts have been made to implement the goal programming over QFD analysis. It is important to note that there is no model that implements multi-objective optimization approach over QFD analysis.

In this chapter, we proposed the DMOEA algorithm to resolve PTRs selection problem in product planning. The proposed algorithm is tested over the case study of software development. The result shows that the DMOEA is highly capable to handle the conflicts amongst the objectives, namely FANP-QFD, NPD cost, manufacturability, NPD time, and Technological advances.

References

Akao Y (1990) Quality function deployment: integrating customer requirements into product design. Productivity Press, Cambridge

Armacost RL, Componation PJ, Mullens MA, Swart WW (1994) An AHP framework for prioritizing customer requirements in QFD: an industrialized housing application. IIE Trans 26(4):72–79

Buyukozkan G, Kahraman C, Ruan D (2004) A fuzzy multi-criteria decision approaches for software development strategy selection. Int J Gen Syst 33(2–3):259–280

Chan LK, Kao HP, Ng A, Wu ML (1999) Rating the importance of customer needs in quality function deployment by fuzzy and entropy methods. Int J Prod Res 37(11):2499–2518
Chen Y, Tang J, Fung RYK, Ren Z (2004) Fuzzy regression-based mathematical programming model for quality function deployment. Int J Prod Res 42(5):1009–1027
Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans Evol Comput 6(2):182–197
Eiben AE, Raue PE, Ruttkay ZS (1994) Genetic algorithms with multi-parent recombination. Proceeding of the 3rd conference on parallel problem solving from nature, LNCS 866, Springer-Verlag, pp 78–87
Einhorn HJ, Hogarth RM (1986) Decision making under ambiguity. J Bus 59:225–250
Ertay T, Buyukozkan G, Kahraman C, Ruan D (2005) Quality function deployment implementation based on analytic network process with linguistic data: an application in automotive industry. J Intell Fuzzy Syst 16(1):221–232
Fleming P, Purshouse RC, Lygoe RJ (2005) Many objective optimization: an engineering design perspective. In: Evolutionary multi-criterion optimization, LNCS, vol 3410, pp 14–32
Fleming P, Purshouse RC (2007) On the evolutionary optimization of many conflicting objectives. IEEE Trans Evol Comput 11(7):770–784
Fonseca CM, Fleming P (1993) Genetic algorithms for multi-objective optimization: Formulation, discussion and generalization: In: Forrest S (ed) Proceedings of the fifth international conference on genetic algorithms. Morgan Kauffman, San Mateo, pp 416–423
Fukuda S, Matsuura Y (1993) Prioritizing the customer's requirements by AHP for concurrent design. ASME Des Eng Div 52:13–19
Griffin A, Hauser JR (1993) The voice of customer. J Mark Sci 12(1):1–27
Hauser JR, Clausing D (1988) The house of quality. Harvard Bus Rev 66:63–73
Ho ES, Lai YJ, Chang SI (1999) An integrated group decision-making approach to quality function deployment. IIE Trans 31(6):553–567
Kahraman C, Ertay T, Buyukozkan G (2006) A fuzzy optimisation model for QFD planning process using analytic network approach. Eur J Oper Res 171(2):390–411
Karsak EE, Sozer S, Alptekin SE (2002) Product planning in quality function deployment using a combined analytic network process and goal programming approach. Comput Ind Eng 44(1):171–190
Kim KJ, Moskowitz H, Dhingra A, Evans G (2000) Fuzzy multi-criteria models for quality function deployment. Eur J Oper Res 121(3):504–518
Klir GJ, Yuan B (1995) Fuzzy sets and fuzzy logic theory and applications. Prentice-Hall International, London
Knowles J, Corne D (1999) The Pareto archived evolution strategy: a new baseline algorithm for Pareto multi-objective optimization. Proceeding of the 1999 congress on evolutionary computation, Washington, DC, USA
Koppen M, Yoshida K (2007) Substitute distance assignments in NSGA-II for handling multi-objective optimization problems. In Proceedings of 4th international conference evolution multi-criteria optimization, vol 4403. Springer-Verlag, New York, pp 727–741
Kwong CK, Bai H (2002) A fuzzy AHP approach to the determination of importance weights of customer requirements in quality function deployment. J Intell Manuf 13:367–377
Lee AHI, Chen HH, Tong Y (2008) Developing new products in a network with efficiency and innovation. Int J Prod Res 48(17):4687–4707
Lee AHI, Kang HY, Yang CY, Lin CY (2009) An evaluation framework for product planning using FANP, QFD and multi-choice goal programming. Int J Prod Res 48(13):3977–3997
Lu M, Madu CN, Kuei C, Winokur D (1994) Integrating QFD, AHP, and benchmarking in strategic marketing. J Busi Indust Market 9(1):41–50
Park T, Kim K (1998) Determination of an optimal set of design requirements using house of quality. J Oper Manage 16(5):569–581
Saaty TL (1980) The analytic hierarchy process. McGraw-Hill, New York
Saaty TL (1996) Decision making with dependence and feedback: the analytic network process. RWS Publications, Pittsburgh

Shen XX, Tan KC, Xie M (2001) The implementation of quality function deployment based on linguistic data. J Intell Manuf 12(1):65–75

Shillito ML (1994) Advanced QFD: linking technology to market and company needs. Wiley, New York

Sakawa M, Yano H, Yumine T (1987) An interactive fuzzy satisfying method for multi-objective linear programming problems and its application. IEEE Trans Syst Man Cyber-Part A: Syst Hum 17(4):654–661

Yager RR (1978) On a general class of fuzzy connective. Fuzzy Sets Syst 4(3):235–242

Zitzler E, Laumanns M, Thiele L (2001) SPEA2: improving the strength Pareto evolutionary algorithm. Technical report, TIK-report 103, Swiss Federal Institute of Technology, Lausanne, Switzerland

Zou X, Chen Y, Liu M, Kang L (2008) A new evolutionary algorithm for solving many-objective optimization problems. IEEE Trans Syst Man Cyber-Part B 38(5):1402–1412

Chapter 5
Multi-objective Ant Colony Optimization Method to Solve Container Terminal Problem

F. Belmecheri-Yalaoui, F. Yalaoui and L. Amodeo

Abstract The river and maritime transport represents an attractive alternative to land and air transport. The containerization allows the industries to save costs thanks to the standardization of dimensions. The container terminal has to manage container traffic at the crossroads of land road and railway. In this chapter, we propose to optimize, simultaneously, the storage problem and the quayside transport problem. In a space storage, we have several blocks and each one has its storage cost. The first aim is to minimize the cost storage of containers. These latter are loaded into vessels, the vehicles have to transport the containers from blocks to quays (of vessels). Thus, the second aim consists to minimize the distance between the space storage and the quays. The optimization methods of operations research in container terminal operation have become more and more important in recent years. Objective methods are necessary to support decisions. To solve this multi-objective problem, we develop two resolution methods based on metaheuristic approach called ant colony algorithm. The first one is multi-objective ant colony optimization (noted MOACO) and the second one is the MOACO with a local search (called MOACO-LS), good promising results are given.

Keywords Multi-objective ant colony method · Vehicle routing problem · Storage problem · Terminal container

F. Belmecheri-Yalaoui
Paris Descartes University, 143 avenue de Versaille, 75016 Paris, France

F. Yalaoui (✉) · L. Amodeo
Charles Delaunay Institute-OSI, University of Technology of Troyes, 12 rue marie curie 10010 Troyes, France
e-mail: farouk.yalaoui@utt.fr

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_5,

5.1 Introduction

One of the most used transportations is the river and maritime transport. It represents an attractive alternative to land and air transport. Indeed, different industries (companies) try several opportunities for reducing costs of this transport. First adopted solution is the containerization which became widespread to river and maritime transport of goods. This solution allows the industries to save costs thanks to the standardization of dimensions, the security of goods, and the fast handling operations for loading/unloading.

However, container can induce to high investment costs in storage, supply, and maintenance of equipment's. The seaport has to manage container traffic at the crossroads of land road, railway. It has also a role for managing: a stock of containers on its site, and a platform of quayside. These managements solve a set of decision problems, and the aim is to reduce the costs of transport.

Generally, container terminals can be described as open systems of material flow with two external interfaces (Fig. 5.1). The first ones are the quays (or quayside) with loading and unloading of vessels (or ships). The second ones are the landside where containers are loaded and unloaded on trucks and trains. Containers are stored in stacks thus facilitating the decoupling of quayside and landside operation. When containers, transported by a container vessel, are arrived at the port, they are assigned to a berth with cranes for loading and unloading. Containers arrived by routes (road) or railway at the terminal are handled by using the trucks (or by train). They are transferred to the respective blocks (or stacks) in the zone (yard). Sometimes, we can use the waterside transshipment process term to define the movement to and from the quay. The container storage area is ordinary composed into different blocks characterized by rows, bays, and tiers. There are specific containers: containing dangerous goods, needing electrical connection. Often, blocks are composed into areas for export, import, special, and empty containers. Besides in these general functions some terminals differ also in their operational units.

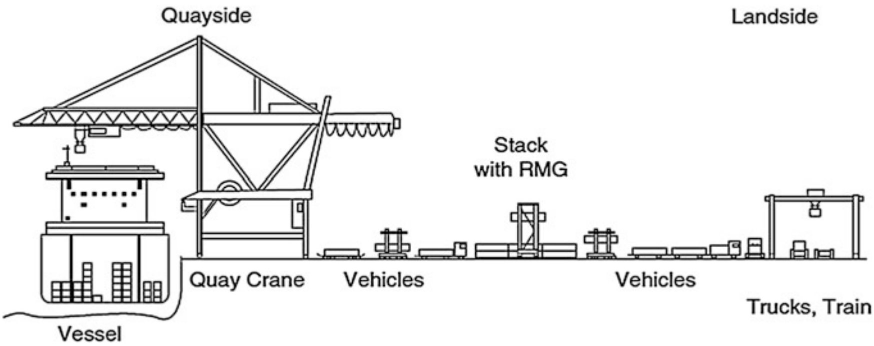


Fig. 5.1 Container terminal system (schematic side view, not true to size) Steenken et al. (2004)

The optimization methods of operations research in container terminal operation have become more and more important in recent years. Objective methods are necessary to support decisions. Here, some works for each problem are exposed. For the berth allocation problem, each berth is allocated to each vessel (or ship), the ship planning is known one year in advance. We can find the works of Li et al. (1998). They solved a more general problem "scheduling with multiple-job-on-one-processor pattern" where the aim is to minimize the makespan of the schedule where vessels represent jobs. A heuristic approach based on mixed-integer programming is proposed by Imai et al. (2003) to solve the berth allocation. Park and Kim (2003) use an approach for assigning berth with quay crane capacities.

Concerning the storage allocation problem, containers are stacked in several levels and the whole storage area is separated into blocks. The maximum number of levels depends on the stacking equipment, either straddle carriers or gantry cranes. Kim and Kim (1998, 1999a, 2002 and 2003) calculated the optimal amount of storage space and optimal number of cranes for containers. The authors used the fixed investments costs and variable operation costs. The objectives are the minimization of the costs of only the terminal operator and minimization of these costs combined with the costs of the customers. Kim and Park (2003) proposed two heuristic algorithms to outbound containers arriving at a storage yard and a subgradient optimization technique. The storage space allocation problem is also studied by Zhang et al. (2003). Their problem is decomposed into two levels. The workload among blocks is balanced at the first level. The total number of containers associated with each vessel and allocated to each block is a result of the second step which minimizes the total distance to transport containers between blocks and vessels. Additional references for storage and stacking logistics are, e.g., De Castilho and Daganzo (1993), Holguin-Veras and Jara-Diaz (1999), Kim and Kim (1999a, b, c) and Kozan and Preston (1999).

For the transport problem, we distinguish two types of transport at a container terminal: the quayside transport and landside transport (see Belmecheri et al. 2009). In the quayside transport, the vessels are loaded/unloaded by the containers which are transported from storage blocks to quaysides. Transport optimization at the quayside, not only means to reduce transport times, but also to synchronize the transports with the loading and unloading activity of the quay cranes. Among the aims of the transport optimization, we can find the goal to reduce the transport times, distance, and also to synchronize the transports with the loading and unloading activity of the quay cranes. Different modes of transport and strategies to allocate vehicles like Automatic Guided Vehicle (AGV) to cranes occur at the quayside. In single-cycle mode the vehicles serve only one crane. According to the crane's cycle they either transport discharged containers from the quay to the yard or export containers from the yard to the crane. The objective of optimization in any case is to minimize the lateness of container deliveries for the cranes, distance, and the travel times of the transport vehicles. Evers and Koppers (1996) develop a model of an AGV traffic control system. The rules for management of empty AGV are developed by Van der Heijden et al. (2002). The authors Kim and Kim (1999c) minimized the total travel distance of straddle carriers in the yard. They propose a beam search algorithm.

The remainder of the chapter is organized as follows. Section 5.2 describes the studied problem. The process of development of the multi-objective ant colony optimization (MOACO) algorithm is defined in Sect. 5.3. Section 5.4 is devoted to a computational evaluation in which MOACO and MOACO with a local search, called MOACO-LS, are compared on several instances. Section 5.5 concludes the chapter.

5.2 Problem Description

In this study, we assume only one planning horizon. We consider during a certain number of days, some containers of customers (industries) are received by the container terminal (maritime port), they are placed in the storage space then they are loaded in vessels (Fig. 5.2).

Notations	
I	Number of customers
i	Customer index: $i = \{1, 2..I\}$
B	Total number of blocks in the storage space
b	Block index: $b = \{1, 2\ldots B\}$
Q	Total number of quays
q	Quay index: $q = \{1, 2\ldots Q\}$
$\text{cap}B_b$	Storage capacity of block b
$\text{cap}Q_q$	Capacity of quay q
co_i	Number of containers of customer i
c_b	Storage cost of block b
d_{bq}	Distance from block b to quay q
d_{bq}^{inv}	Distance from quay q to block b
$x_{ib} \in \{0, 1\}$	$x_{ib} = 1$ if the containers of customer i are stored in block b.

We assume I customers, and each customer i wants to send some containers co_i by the maritime transport. The study includes a container port composed to storage space allocation and a set of quaysides. The storage space has several storage blocks b where $b \in [1, 2\ldots B]$. For each block b, a cost of storage container is noted c_b. When the containers are assigned to blocks, they are loaded into vessels. co_i have to be transported from b to quay $q(q \in [1, 2\ldots Q])$. The distance between blocks and quays is noted d_{bq} (reverse distance is d_{bq}^{inv}).

Our objectives are:

- The minimization of the cost storage.
- The minimization of total traveled distance by vehicles between blocks storage and vessels.

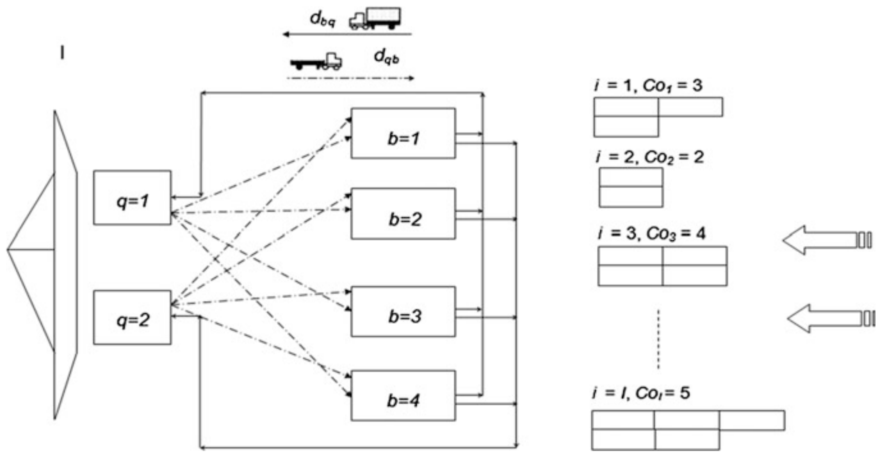


Fig. 5.2 Problem description

5.3 Resolution Methods

To solve this multi-objective problem, we develop a resolution method based on metaheuristic approach called Ant colony algorithm. First, the multi-objective ant colony optimization is presented in Sect. 5.3.1 and then we explain, in Sect. 5.3.2, its adaptation to tackle the problem.

5.3.1 Multi-objective Ant Colony Optimization

In general, an ant colony optimization is a technique to solve difficult combinatorial problems. It is inspired by the behavior of ant colonies. They cooperate to find good paths through graphs for finding the best path from nest to food. The artificial ants seek the solutions according to a constructive procedure. The construction of solutions is based on a probabilistic technique. The solutions are guided by (artificial) pheromone trails which change in a dynamic way (at each iteration) and a heuristic information.

Recently, this metaheuristic is extended to deal with multi-objective optimization problems. Indeed, multi-objective Ant colony optimization (MOACO) has proved its effectiveness in several works. Gambardella et al. (1999) solved the vehicle routing problem with time windows by multiple ant colony system (Doerner et al. 2003). In scheduling problem, Gravel et al. (2002) proposed a MOACO. T'kindt et al. (2002) present an ant colony optimization to solve bi-objective flow shop scheduling problems. MOACO is even used in combinatorial problems as the works of Dugardin et al. (2009).

5.3.2 Adapted MOACO

Since the first work of Dorigo et al. (1996), several versions of ant-based algorithms developed to solve combinatorial problems. Here, we use the Ant Colony System. It is based on three main steps: the solutions encoding, the ants tours construction, the global pheromone updates. But in this case, the number of pheromone matrices is more than one. The multi-objective optimization takes different criterion simultaneously. So, the number of pheromone matrices is equal to the number of objectives to be optimized. Therefore, two pheromone matrices are used in our works. Two colonies of ants collaborate mutually between them: the first one is called ACO1 for storage cost optimization; the second is ACO2 for distance traveled optimization. ACO1 communicates with ACO2 with the heuristic information and the pheromone trails to build solutions.

5.3.2.1 Solution Encoding

Figure 5.3 is a representation of two solutions obtained by two ants. We consider 5 customers, 3 blocks, and 3 quays. In the first part, we would to find a good solution in assigning each customer to each block. We can observe a partial solution of two ants. For ant 1 (ant 2), the containers of customer 2 are stored in block 3 (block 1). Then they are transported to quay 1 (quay 2). So, the ant 1 (ant 2) has found the cost, of maritime transport for customer 2, equal to 12 (equal to 11.8).

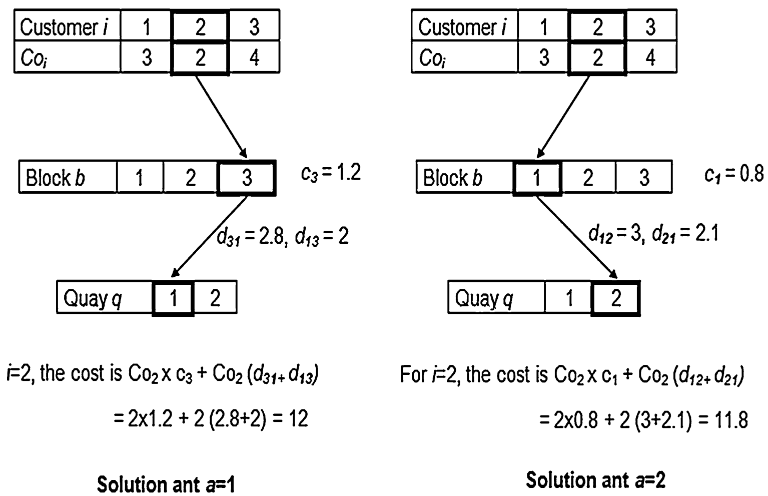


Fig. 5.3 Example of solutions construction (2 ants, 3 customers, 3 blocks, 2 quays)

5.3.2.2 Solution Construction for Storage Cost Optimization

The first colony called ACO1 (Ant Colony Optimization 1) affects the containers in the space storage. Each ant is deposited randomly on a point. This point represents the demand of one customer which is a set of containers. This set is selected to be assigned for one block of storage. The points to be visited are chosen based on the application of the state transition rules. An ant a chooses a customer i for storing its containers co_i to block b.

$$P1_{ib} = \left\{ \frac{\eta_i^{\alpha_1} . \tau_{ib}^{\beta_1}}{\sum_j \eta_j^{\alpha_1} . \tau_{jb}^{\beta_1}} \quad \forall i,j \in \Omega,\ \forall b \in B \right. \tag{5.1}$$

$P1_{ib}$ represents the probability for choosing containers of customer i to be assigned to block b. Ω denotes the customers which are not yet chosen by the current ant. τ_{ib} is the pheromone quantity deposited by ants on node (i, b) which measures the acquired desirability between the customer i and block b. Here, b selected by ant a is determined by Eq. (5.2). η_i is the heuristic information (visibility) of customer i. The parameters α_1, β_1 determine, respectively, the relative importance of pheromone trails and heuristic information to select customer i.

$$\eta_i = \max_{b \in B} [1/(co_i . c_b)] \quad \forall i \in \Omega \tag{5.2}$$

This formula allows to containers of customer i (cited co_i) to be stored in the space allocation with a minimum cost in choosing an adapted block b. Here, we give the ACO1 procedure to optimize the cost storage with the rules and the concepts described above.

Algorithm 5.1: ACO1

Data: B, I, Q
Initialize parameters $\alpha_1, \beta_1, \rho_1$
Initialize τ_{ib}=1,$\forall i \in I, \forall b \in B$
Begin
    $\Omega \leftarrow I$
    While $\Omega \neq \emptyset$ do
        For $i \in \Omega$ do
        Calculate the probability $P1_{ib}$

$$P1_{ib} = \left\{ \frac{\eta_i^{\alpha_1} . \tau_{ib}^{\beta_1}}{\sum_j \eta_j^{\alpha_1} . \tau_{jb}^{\beta_1}} \quad \forall i,j \in \Omega, \forall b \in B \right.$$

        Select randomly a single customer i to be stored in block b
        $\Omega \leftarrow \Omega - \{i\}$
    Solution Sol_a^1 is built
    Calculate the solution cost $Cost_a^1$

5.3.2.3 Solution Construction for Distance Optimization

When the containers are assigned to the storage blocks, we apply the second colony called ACO2 (ant colony optimization 2) to optimize the distance traveled by vehicles for transporting these containers to quays. Each demand of customer is chosen randomly with his probability $P2_{iq}$.

$$P2_{iq} = \left\{ \frac{\delta_i^{\alpha_2}.\upsilon_{iq}^{\beta_2}}{\sum_j \delta_j^{\alpha_2}.\upsilon_{jq}^{\beta_2}} \quad \forall i, j \in \Omega, \ \forall q \in Q \right. \tag{5.3}$$

υ_{iq} is the pheromone quantity deposited by ants on node (i, q) which measures the acquired desirability between the customer i and quay q. Equation (5.4) calculates the visibility of i. It selects the quay q to receive the containers of i. The parameters α_2, β_2 modulate the pheromone trails and heuristic information to select i.

$$\delta_i = \max_{q \in Q} \left[1/\left(co_i.(d_{bq} + d_{bq}^{\text{inv}}) \right) \right] \quad \forall i \in \Omega \tag{5.4}$$

The desirability to select the containers of i to be brought to quay q is calculated with Eq. (5.4).

Algorithm 5.2: ACO2

Data: *B, D, I, Q*
Initialize parameters α_2, β_2, ρ_2
Initialize υ_{iq}=1,$\forall i \in I, \forall q \in Q$
Begin
    $\Omega \leftarrow I$
    While $\Omega \neq \emptyset$ do
        For $i \in \Omega$ do
        Calculate the probability $P2_{iq}$

$$P2_{iq} = \left\{ \frac{\delta_i^{\alpha_2}.\upsilon_{iq}^{\beta_2}}{\sum_j \delta_j^{\alpha_2}.\upsilon_{jq}^{\beta_2}} \quad \forall i, j \in \Omega, \forall q \in Q \right.$$

        Select randomly a single customer i to be transported to quay q
        $\Omega \leftarrow \Omega - \{i\}$
    Solution Sol_a^2 is built
    Calculate the solution cost $Cost_a^2$

5.3.3 *The Proposed Algorithm: MOACO*

The algorithm MOACO for optimizing the cost storage and distance simultaneously is given in Algorithm 5.3. The process is applied T and A times which are respectively the number of iterations and ants. Each ant a constructs one solution

Sol_a^1 for storing the containers of customers I in blocks B with Algorithm 5.1. Then, this solution is evaluated with Cost_a^1 which is equal to $\sum_{i=1}^{I} \sum_{b=1}^{B} x_{ib} co_i c_b$. The second task of ant a concerns the construction of one solution Sol_a^2 for transporting the containers from blocks to quays Q. Its evaluation Cost_a^2 is computed as $\sum_{i=1}^{I} \sum_{b=1}^{B} \sum_{q=1}^{Q} x_{ib} co_i (d_{bq} + d_{bq}^{\text{inv}})$.

After that, we compute the complete solution Cost_a which is a vector with coordinates (Cost_a^1, Cost_a^2). When the ants A have constructed the solutions at iteration t. Each solution can update the new Pareto front F which is the set of non-dominated solutions $F = F_1, F_2 \ldots, F_r, ..F_R$. The F_r, which is initially a non-dominated solution, is composed by coordinate $x(y)$, where $F_r.x = \text{Cost}_a^1$ ($F_r.y = \text{Cost}_a^2$,) and the costs of a was found at an iteration before current t.

Algorithm 5.3: MOACO

Data: B, D, I, Q
Initialize parameters $\alpha_1, \beta_1, \rho_1, \alpha_2, \beta_2, \rho_2, T, A$
Initialize τ_{ib}=1, υ_{iq}=1,$\forall i \in I, \forall b \in B, \forall q \in Q$
Begin
    For $t \in T$ do
        For $a \in A$ do
            // Construct a complete solution
            Part 1
            Construct storage solution with ACO1 procedure $Sol_a{}^1$
            Evaluate storage solution $Cost_a{}^1$

            Part 2
            Construct storage solution with ACO2 procedure $Sol_a{}^2$
            Evaluate storage solution $Cost_a{}^2$
            Evaluate complete solution $Cost_a$

        Construct the non-dominated solutions of ants A
        Update the different fronts F_r of colony (algorithm 5.4)
        Update the fronts with current and preceding iterations of the new best front
        Update pheromone for storage solutions (Eq. 5.5, 5.6)
        Update pheromone for distance solutions (Eq. 5.7, 5.8)
    Return the best Pareto front

The aim is to find new non-dominated solutions obtained by colony at iteration t, then the front is updated by Algorithm 5.4. Each non-dominated solution called NDS of ant a is compared with solutions front F_r for $r \in R$. When NDS is non-dominated by at least one solution front then Pareto front is updated with NDS of a.

Algorithm 5.4: Update Pareto front

```
Begin
    For a ∈ A do
        If solution of a is NDS by the request of other ants then
            CheckNDS=false
            Nonstop=true
            r=1
            While (r ≤ R or ¬CheckNDS) and Nonstop do
                If Cost_a^1 ≤ F_r.x and Cost_a^2 ≤ F_r.y then
                    CheckNDS=true
                Else
                    CheckNDS=false
                If (r ≥ R and CheckNDS) then
                    Nonstop=false
            r=r+1
        If CheckNDS then
        Update the Pareto front
```

5.3.4 Pheromone Update (Storage and Distance Part)

When ants have built solutions (stored and transported containers), the pheromone update process is applied to find the best Pareto front. For the storage part, the solutions having arcs (*i, b*) in the Pareto front are updated as follows:

$$\tau_{ib} = (1 - \rho_1).\tau_{ib} + \rho_1.\Delta^1_{ib} \tag{5.5}$$

where

$$\Delta^1_{ib} = \begin{cases} Cost^1_a / F_r.x & \text{if } \exists (i,b) \in Sol(F_r), \forall r \in R \\ 0 & \text{otherwise} \end{cases} \tag{5.6}$$

The trail on pheromone deposited on path (*i, b*) will be increased if there is at least one non-dominated solution F_r of front Pareto *F*. ρ_1 $(0 < \rho_1 < 1)$ is the evaporation rate used to explore more the search space. The same process is applied on distance optimization part. The pheromone levels of the solution components (*i, q*) existing in the Pareto front are updated by Eq. (5.7).

$$\upsilon_{iq} = (1 - \rho_2).\upsilon_{iq} + \rho_2.\Delta^2_{iq} \tag{5.7}$$

where

$$\Delta^2_{iq} = \begin{cases} Cost^2_a / F_r.y & \text{if } \exists (i,q) \in Sol(F_r), \forall r \in R \\ 0 & \text{otherwise} \end{cases} \tag{5.8}$$

ρ_2 is the evaporation rate for exploring the new search space. The amount quantity pheromone added Δ_{iq}^2 is calculated thanks to Eq. 5.8. If arc (i, q) is traveled in at least one dominated solution F_r of Pareto front F then the quantity added depends on the cost of the second part and the cost of $F_r.x$.

5.3.5 MOACO-LS

The second proposed approach is MOACO with a local search, called MOACO-LS. This one is applied on non-dominated solutions found after some iteration. The moves consist to relocate a set of containers of customer i with those of customer j in checking the capacity of block and quay containing co_i and the block and quay having co_j. The aim is to verify the behavior of MOACO when the local search is added.

5.4 Computational Results

To analyze the solution quality of the proposed algorithms, several experiments are conducted on generated data. The tests are applied with MOACO approach and MOACO-LS approach. The parameters values of these approaches are determined thanks to several experiences. To evaluate the obtained Pareto front, three comparison criteria are used: the number of solutions in an optimal front, the distance proposed by Riise (2002), and the Zitlzler measure (Zitzler and Thiele 1999).

5.4.1 Input Data

To confirm if the both approaches return high quality solutions, we develop a complete enumeration which is applied on small data. So, we generate randomly the data of size $I = 5$ and $I = 10$. As we have said above, we consider a single planning horizon. We assume that during one day, we receive some containers which will be stored in the blocks then assigned to vessels devoted to one country. For 5 customers, we assume 3 blocks and 2 quays (devoted to 1 vessel). For 10 customers, we have 5 blocks and 4 quays (2 quays per vessel). Each customer i has a set of containers co_i generated randomly in [1, 7]. Each block has its capacity and its cost storage which are generated randomly in [4, 20] and [2.00, 4.00], respectively. Each quay has a defined distance with each block.

5.4.2 Evaluation Metrics

Here, we assume *F1* and *F2* which are the Pareto front of method 1 and method 2 respectively. To compare the quality solutions of these Pareto fronts, three criteria are used. The first one is the number of solutions *nF1* (*nF2*) in an optimal front *F1* (*F2*).

The second one is the distance of Riise μ calculated with Eq. 5.9, d_s is a distance between solution *s* belonging to front *F2* and its orthogonal projection on front *F1*. When d_s has a negative (positive) value, it means that *F2* is under (upper) *F1*. The μ value depends to the number of solutions *nF* in each front *F*, a normalized value is often used (Eq. 5.10).

$$\mu = \sum_{s=1}^{nF} d_s \tag{5.9}$$

$$\mu^* = \frac{\mu}{nF} \tag{5.10}$$

The last comparison criterion is Zitzler measure: *C (F1, F2), C (F2, F1). C (F1, F2) (C (F2, F1))* represents the percentage of solutions in *F1 (F2)* dominated by at least one solutions of *F2 (F1)*. The front *F1* is better than front *F2* if *C (F1, F2)* $<$ *C (F2, F1)*.

5.4.3 Parameter Setting

To compare the two algorithms, several tests are made to evaluate the good values of different parameters. Final values are determined as a compromise between the quality of the final solutions and the convergence time needed. Those parameters have been chosen based on a set of experimental tests. In each test, we tried to determine the best value for a parameter while fixing the other ones. The value parameters are $\alpha_1 = 0.5$, $\beta_1 = 0.1$, $\rho_1 = 0.7$, $\alpha_2 = 0.7$, $\beta_2 = 0.15$, $\rho_2 = 0.8$.

At the beginning of the algorithms, the parameters values are initiated to $\tau_{ib} = 1\ \forall i \in I$ and $\forall b \in B$, $\upsilon_{iq} = 1\ \forall i \in I$ and $\forall q \in Q$. Concerning the stopping criterion, they are same for both (MOACO, MOACO-LS). For $I = 5$ ($I = 10$), the number of iterations is $T = 100$ ($T = 1000$) and the number of ants is $A = 5$ ($A = 10$). About MOACO-LS, the Local Search (LS) is called after 10 (100) iterations for $I = 5$ ($I = 10$). To note that the local search is applied on the non-dominated solutions found. It means that the LS is applied 10 (100) times for $I = 5$ ($I = 10$).

5.4.4 Analysis

To evaluate the convergence of the approaches toward optimal solutions and to confirm if the obtained solutions are a good quality, we compare in Tables 5.1 and 5.2 the results of MOACO and MOACO-LS with those of full Enumeration Method (EM). The instance used is $I = 5$ because the EM induces to an important computing time.

In Tables 5.1 and 5.2, the results show, based on the 5 tested instances, that the MOACO converge to optimal solution in all cases and the MOACO-LS converge to optimal solutions in 4 instances and the 1 other is very close to optimal solutions. Here, the *nF1* and *nF2* represent the number of solutions of fronts *F1* and *F2*, respectively. The distance of Riise is noted as μ. Finally, *C (F1, F2)* and *C (F2, F1)* denote the Zitzler measure. In referring to these tables, we can confirm the advantage of developing the MOACO and MOACO-LS which are able to get optimal solutions for several instances.

The last comparison, in Table 5.3, is between the MOACO and the MOACO-LS algorithms on the generated instance of size 10 customers. We can see the comparison of two Pareto fronts obtained by algorithm MOACO (noted *F1*) and algorithm MOACO-LS (noted *F2*). The Fig. 5.4 shows the Pareto fronts obtained by MOACO and MOACO-LS on one tested instance.

We give an example for reading the Table 5.3, in the first line (instance Pb10-1), the number of non-dominated solutions *nF1* in the MOACO Pareto front is 2 instead of 3 for the MOACO-LS Pareto front. The negative distance of Riise $\mu = -0.68$ and $\mu^* = -0.23$ indicate that MOACO-LS front is under the MOACO

Table 5.1 Comparison between EM (*F1*) and MOACO (*F2*) algorithms

Instance	*nF1*	*nF2*	μ	μ*	*C (F1, F2)*	*C (F2, F1)*
Pb5-1	1	1	0	0	0	0
Pb5-2	1	1	0	0	0	0
Pb5-3	1	1	0	0	0	0
Pb5-4	1	1	0	0	0	1
Pb5-5	2	2	0	0	0	0
Average	1.2	1.2	0	0	0	0.2

Table 5.2 Comparison between EM (*F1*) and MOACO-LS (*F2*) algorithms

Instance	*nF1*	*nF2*	μ	μ*	*C (F1, F2)*	*C (F2, F1)*
Pb5-1	1	1	0	0	0	0
Pb5-2	1	1	0	0	0	0
Pb5-3	1	1	0	0	0	0
Pb5-4	1	1	0	0	0	0
Pb5-5	2	2	0	0	0	0
Average	1.2	1.2	0	0	0	0

Table 5.3 Comparison between MOACO (F1) and MOACO-LS (F2) algorithms

Instance	*nF1*	*nF2*	μ	μ^*	*C (F1, F2)*	*C (F2, F1)*
Pb10-1	2	3	−0.68	−0.23	0	0
Pb10-2	3	3	16.52	5.51	0	0.33
Pb10-3	3	4	−12.36	−3.09	0.67	0
Pb10-4	3	3	4.85	1.62	0.33	0
Pb10-5	2	3	−5.26	−1.75	0.5	0
Pb10-6	3	4	−13.9	−3.47	0.67	0.75
Pb10-7	4	2	−38.72	−19.36	0	1
Pb10-8	4	3	−7.69	−2.56	0.25	1
Pb10-9	4	3	−9.21	−3.07	0.25	0.33
Pb10-10	3	4	2.84	0.71	0.33	0.25
Pb10-11	2	3	−60.34	−20.11	0.5	0.67
Pb10-12	3	2	10.52	5.26	0.67	0
Pb10-13	2	3	−71.33	−23.78	0	0.33
Pb10-14	2	3	−8.27	−2.76	0	0.33
Pb10-15	3	2	−0.27	−0.13	0.67	0
Pb10-16	1	1	0	0	0	0
Pb10-17	2	4	17.93	4.48	1	0
Pb10-18	4	1	5.54	5.54	0.5	0
Pb10-19	3	3	0	0	0	0.33
Pb10-20	5	9	−10.1	−1.12	0.4	0.22
Pb10-21	1	1	0	0	0	0
Pb10-22	2	2	−11.73	−5.86	0	1
Pb10-23	2	1	2.81	2.81	0.5	0
Pb10-24	2	3	10.01	3.34	0	0.33
Pb10-25	2	3	−6.08	−2.03	0	0.33
Pb10-26	4	2	−3.15	−1.57	0.25	0
Pb10-27	1	1	5.13	5.13	0	0
Pb10-28	1	1	8.05	8.05	1	0
Pb10-29	5	2	12.88	6.44	0.2	0
Pb10-30	8	4	−77.14	−19.28	0.38	0.5
Average	2.8	2.77	−7.97	−2.04	0.32	0.26

front, it means that on this case the MOACO-LS is better than MOACO. In Pb10-3, 67 % of solutions obtained by MOACO are dominated by at least one solution of those of the MOACO-LS while no solution of MOACO-LS front is dominated by the MOACO front.

In referring to these results, we can conclude that in several generated instances, the MOACO-LS is more interesting compared to the MOACO algorithm. The average value of the Riise distances μ is −7.97, it means that the MOACO-LS Pareto front is under the MOACO Pareto front. Concerning the Zitzler measure, we can see that the average *C (F1, F2)* is 0.32 which represents that 32 % of solutions in the MOACO Pareto front are dominated by at least one solution from the MOACO-LS front.

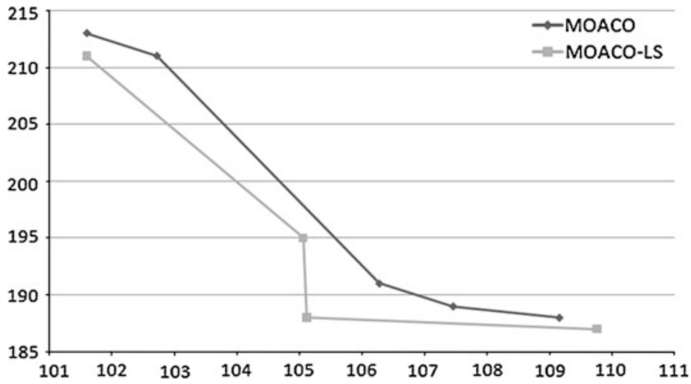


Fig. 5.4 Pareto fronts on one tested instance

5.5 Conclusion

We have studied a multi-objective container terminal problem with an ant colony optimization. The two objectives are the minimization of cost storage and the minimization of distance transport. The multi-objective metaheuristics developed are a MOACO and a MOACO-LS. The tests are applied on several instances. We have compared the results obtained by the two approaches with a complete enumeration. We have noticed that the MOACO-LS is better than MOACO. In the perspectives of this work, other methods based on the Pareto could be developed such as a genetic algorithm.

References

Belmecheri F, Cagniart T, Amodeo L, Yalaoui F (2009) Modelling and optimization of empty container reuse: a real case study. In: IEEE proceedings of the international conference of computers and industrial engineering, Troyes, pp 1106–1109

De Castilho B, Daganzo CF (1993) Handling strategies for import containers at marine terminals. Transp Res B 27(2):151–166

Doener K, Hartl RF, Teimann M (2003) Are competants more competent for problem solving? The case of full truckload transportation. Eur J Oper Res 11(2):79–99

Dorigo M, Maniezzo V, Colorni A (1996) The ant system: optimization by a colony of cooperating agents. IEEE Trans Syst Man Cybern Part B 26(1):29–41

Dugardin F, Yalaoui F, Amodeo L (2009) New multi-objective method to solve reentrant hybrid flow shop scheduling problem. Eur J Oper Res 203(1):22–31

Evers JJM, Koppers SAJ (1996) Automated guided vehicle traffic control at a container terminal. Transp Res A 30(1):21–34

Gambardella L, Taillard E, Agazzi G (1999) MACS-VRPTW: a multiple ant colony system for vehicle routing problems with time windows. Orne D, Dorigo M, Glover F (eds) McGraw-Hill, New York

Gravel M, Price WL, Gagn C (2002) Scheduling continuous casting of aluminum using a multiple objective ant colony optimization metaheuristic. Eur J Oper Res 143(1):218–229

Holguin-Veras J, Jara-Diaz S (1999) Optimal pricing for priority service and space allocation in container ports. Transp Res-B 33(3):81–106

Imai A, Nishimura E, Papadimitriou S (2003) Berth allocation with service priority. Transp Res B 37(5):437–457

Kim KH, Kim HB (1998) The optimal determination of the space requirement and the number of transfer cranes for import containers. Comput Ind Eng 35(3–4):427–430

Kim KH, Kim HB (1999a) Segregating space allocation models for container inventories in port container terminals. Int J Prod Econ 59(1–3):415–423

Kim KH, Kim KY (1999b) An optimal routing algorithm for a transfer crane in port container terminals. Transp Sci 33(1):17–33

Kim KH, Kim KY (1999c) Routing straddle carriers for the loading operation of containers using a beam search algorithm. Comput Ind Eng 36(1):106–136

Kim KH, Kim HB (2002) The optimal sizing of the storage space and handling facilities for import containers. Transp Res B 36(9):821–835

Kim KH, Park KT (2003) A note on a dynamic space-allocation method for outbound containers. Eur J Oper Res 148(1):92–101

Kozan E, Preston P (1999) Genetic algorithms to schedule container transfers at multimodal terminals. Int Trans Oper Res 6(3):311–329

Li CL, Cai X, Lee CY (1998) Scheduling with multiple-job-on-one-processor pattern. IIE Trans 30:433–445

Park YM, Kim KH (2003) A scheduling method for berth and quay cranes. OR Spectr 25(1):1–23

Riise A (2002). Comparing genetic algorithm and tabu search for multi-objective optimization. In: IFORS conference, Edinburgh-UK

Steenken D, Vob S, Stahlbock, R (2004) Container terminal operation and operations research- A Classification and Literature Review. Spectrum 26(1):3–49

T'kindt V, Monmarch N, Tercinet F, Lagt D (2002) An ant colony optimization algorithm to solve a 2-machine bicriteria flow shop scheduling problem. Eur J Oper Res 142(2):250–257

Van der Heijden M, Ebben M, Gademann N, Van Harten A (2002) Scheduling vehicles in automated transportation systems: algorithms and case study. OR Spectrum 24:31–58

Zitzler E, Thiele L (1999) Multi-objective evolutionary algorithms: a comparative case study and the strength pareto approach. IEEE Trans Evol Comput 3(4):257–271

Zhang C, Liu J, Wan YW, Murty KG, Linn RJ (2003) Storage space allocation in container terminal. Transp Res B 37(10):883–903

Chapter 6
Exploratory Study in Determining the Importance of Key Criteria in Mobile Supply Chain Management Adoption for Manufacturing Firms: A Multi-criteria Approach

A. Y. L. Chong, F. T. S. Chan and K. B. Ooi

Abstract Mobile supply chain management can help manufacturers to reduce cost and improve supply chain performances. However, the decisions to adopt mobile supply chain management are complex as it involved multi-criterion decisions that need to be considered by manufacturing firms. This research aims to predict the factors that can lead to successful mobile supply chain management adoption. Variables from the technology-organization-environment (TOE) model were used as predictors for this research. A non-compensatory adoption decision process is modeled using neural network analysis. Data was collected from 192 manufacturing firms. Our results showed that some of the strongest predictors for mobile supply chain management adoption are senior management support, security perceptions, technology integrations, and financial and technical competence. Firm size and environmental factors on the other hand have less predictive power than technological and organizational factors on mobile supply chain management adoption decisions.

Keywords Mobile supply chain management · Neural network · Manufacturing firms · Technology adoption decisions

---

A. Y. L. Chong
Nottingham University Business School China, University of Nottingham, Ningbo, China

F. T. S. Chan (✉)
Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, Hong Kong
e-mail: f.chan@inet.polyu.edu.hk

K. B. Ooi
Chancellery Division, Linton University College, Mantin, Malaysia

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_6,

6.1 Introduction

Organizations today operate in a competitive and globalized business environment. Manufacturing organizations in particular, are facing with the pressure of time and quality-based competitions and global sourcing (Chong et al. 2009). Many manufacturing firms have sought to stay ahead of their rivals by offshoring their manufacturing sites. However, employing low cost strategies is not a long-term, sustainable business model. Instead, many successful manufacturing firms have stayed competitive by operating more efficiently through the implementation of supply chain management. One way to improve the implementation of supply chain management is through information systems. In the past, manufacturing firms have relatively good success in implementing information systems in their supply chain. Many manufacturing firms in particular, make use of Internet technologies to implement Electronic Data Interchange (EDI) and Business to Business (B2B) systems (Chou et al. 2004). The applications of Internet technologies help supply chain members cooperate and share information in real time (Gunasekaran and Ngai 2004). By sharing real time information, it helps to reduce supply chain problem such as the bullwhip effect and reduce uncertainties in the supply chain (Yao et al. 2007). Although Web-based applications such as B2B and EDI have help manufacturing firms to improve their supply chain management, one growing technology that has captured the attentions of researchers and practitioners are mobile technologies.

The applications of mobile technologies to supply chain have created a term known as mobile supply chain management. In a mobile supply chain management environment, organizations apply technologies such as radio frequency identification (RFID), Wi-Fi, and GPS to help conduct supply chain activities (Eng 2006). The advantage of applying mobile technologies is that firms will no longer face physical constraints such as the arrangement of cables, and data can be transmitted anytime, anywhere. "Internet of things" is a network that allows organizations to track their products through the supply chain globally, and run multiple applications simultaneously (Ngai et al. 2008). Although mobile supply chain management has many advantages compared to existing technologies, its implementations in manufacturing firms are still relatively low (Wu and Subramaniam 2011). There is currently limited study on the applications and implementations of mobile supply chain management. The implementation decisions of mobile supply chain management are a complicated process involving various decisions. Such views are supported by prior studies by (Ngai et al. 2008). Organizations can adopt mobile supply chain management via mobile devices, Wi-Fi, etc. Second, in most technology adoption studies, researchers have often employed explanatory statistical techniques such as regression analysis to understand the reasons why firms adopt/not adopt technologies. However, the multi-criteria adoption decisions are often very complex, and regression techniques have often oversimplified the problems by examining the decisions in linear and compensatory models. Under such models, researchers believe that the shortfall in one of the adoption decisions such as security can be compensated by improving another adoption factor such as perceived usefulness (Chiang et al. 2006). However, in organizations, the decisions involved are often non-compensatory. For example, an organization which believes that mobile supply chain management's security is a major obstacle in their adoption decisions are unlikely to be compensated by factors such as the cost of mobile supply chain management. This problem is observed by prior studies conducted by Chiang et al. (2006) and Venkatesh and Goyal (2010). Chiang et al. (2006) in their study on information systems adoption found that linear statistical models are not reliable as they are unable to capture the non-compensatory decision rules. Similarly, Venkatesh and Goyal (2010) also argued that linear models tend to oversimplify the complexities involved in information systems adoption decisions.

In order to bridge the gap in existing literatures, this research has several aims. First, this research aims to predict organizations' mobile supply chain management adoption decisions by employing the technology-organization-environment (TOE) model (Tornatzky and Fleisher, 1990). Second, this research aims to use neural network to examine organization's multi-criterion, nonlinear and non-compensatory adoption decisions.

6.2 Literature Review

6.2.1 Overview of Mobile Supply Chain Management

Mobile supply chain management involves the integration of supply chain software applications with mobile technologies (e.g., mobile smart phones, personal digital assistants, Wi-Fi) to provide a ubiquitous, wireless supply chain environment (Eng 2006). Eng (2006) in his study on mobile supply chain management summarized some of the main advantages of mobile supply chain management which include the ability to report real time events, its ubiquity, and the ability to personalize information. In a mobile supply chain environment, important information can be broadcasted in real time without any delays, and the information are transmitted from both inside and outside the organizations. By transmitting the information in real time, it helps organizations to reduce response lag time, avoid delays in processing transactions, and help ensure accurate forecast of products demand and improve customer service (Eng 2006). The prices of products and services can also be dynamic based on real time events (Eng 2006). Such advantages of mobile supply chain management is summed up by Ming et al. (2008) as the 3As features, which are Anyone, Anywhere, Anytime. The ability to personalize information can be achieved through integrating wireless technology with customer relationship management software. Eng (2006) suggested that it is possible to customize and target certain supply chain information based on the needs and circumstances of the individual customers. Another advantage of mobile supply chain management technologies when compared with barcode for example, is that the supply chain members can track products without the needs for scanning. With information on product location, characteristics and inventory levels transmitted precisely, problems related to manual inventory counting, mis-picking of goods in warehouse, and mistakes in order numbering are eliminated (Wang et al. 2010).

Although mobile supply chain management offers many advantages, there are still limited literatures on the factors influencing organizations' adoption decisions. In order to understand the important criteria involved in such decisions, a review of previous studies are important. The next section presents the review.

6.2.2 Factors Affecting the Adoption of Mobile Supply Chain Management

One of the most commonly applied technology adoption model is Tornatzky and Fleischer's TOE model. The TOE model proposed that an organization's decisions to adopt technologies are influenced by the technological, organizational, and environmental factors. Technological factor is the internal and external technologies that are relevant to the organizations. Organizational factors refer to organizations' characteristics such as their size, financial resources, organization structure, etc. Environmental factor refers to the arena where the organizations conduct their business such as the industry which the organizations are operating, regulatory environments and governmental issues (Chong and Ooi 2008).

The TOE model has been applied in various information systems adoption studies. Zhu et al. (2006) applied the TOE model to understand the assimilation of e-business in organizations. Wu et al. (2011) similarly applied the TOE model to predict the adoptions of RFID in organizations' supply chains. Low et al. (2011) used the TOE model to investigate the factors affecting the adoption of cloud computing. Chong and Chan (2012) similarly applied the TOE model to examine the diffusion of RFID adoptions in the health care industry. All these prior studies supported the importance of TOE in explaining the factors affecting information systems adoption decision. Table 6.1 summarizes previous information systems adoption studies which have applied the TOE framework.

As shown in Table 6.1, TOE have been applied and found useful in different types of information systems adoptions (Wang et al. 2010). However, the specific measures specified within the three key constructs in the TOE model vary based on the types of information technologies studied. Drawing upon previous literatures, this study will build its base model based on TOE, and a manufacturing firm's perceived cost, perceived benefits, technology integrations, security perceptions, and complexity are used to measure the technological attributes. In order to measure organizational attributes, firm size, senior management support, technological competence, financial competence are used. In order to measure

Table 6.1 Summary of existing technology adoption studies

Study	Technology studied	Adoption factors
Zhu et al. (2006)	E-business	Technology: technology readiness, technology integration Organization: firm size, global scope, managerial obstacles Environment: competition intensity, regulatory
Wu and Subramaniam (2011)	RFID	Technology: technology maturity, complexity Organization: IT sophistication, top management support Environment: trading partner power, trading partner readiness
Low et al. (2011)	Cloud computing	Technology: relative advantage, complexity, compatibility Organization: management support, firm size, technology readiness Environment: competitive pressure, trading partner pressure
Chong et al. (2012)	RFID	Technology: relative advantage, complexity, compatibility, cost, security Organization: management support, firm size, Financial resources, technological knowledge Environment: competitive pressure, expectation of market trend
Wang et al. (2010)	RFID	Technology: relative advantage, complexity, compatibility Organization: management support, firm size, Technology Competence Environment: competitive pressure, trading partner pressure, information intensity
Kuan and Chau (2001)	EDI	Technology: perceived direct benefits, perceived indirect benefits Organization: perceived financial cost, perceived technical competence Environment: perceived industry pressure, perceived government pressure
Xu et al. (2004)	E-business	Technology: technology competence Organization: firm size, global scope, enterprise integration Environment: competition intensity, regulatory environment
Sharma et al. (2007)	RFID	Technology: perceived benefits, perceived cost Organization: management support, IS infrastructure and capabilities, financial readiness Environment: perceived standard convergence, perceived privacy
Chang et al. (2007)	Electronic signature	Technology: security protection, system complexity Organization: user involvement, resources, organization size, internal need Environment: vendor support, government policy

environmental attributes, competitive pressure, trading partner pressure, and expectations of market trends are being used. The next section provides discussions on the variables used on the three main constructs.

6.2.3 Technological Factors

Perceived cost, perceived benefits, technology integrations, security perceptions are all used as attributes of technological factors. Before implementing mobile supply chain management, organizations will place great considerations in the perceived cost and the expected benefits of the technology . In particular, organizations will need to conduct return on investment calculations given that they will need to invest in technologies such as mobile devices, RDIF tags, and the setting up of the wireless environment. Perceived benefits is similar to relative advantages proposed by prior studies by Chong and Ooi (2008); Chong et al. (2009), and looked at whether mobile supply chain management can have better advantages when compared to their existing technologies. As mobile supply chain management is not a standalone application, and will need to be integrated to organizations' existing systems such as their accounting system, customer relationship management software, etc., the complexity of integrating mobile supply chain management applications into their existing business processes will be an important decision. As stated by Zhu et al. (2006), integration of systems involving back office information systems and database, as well as those external integrated with suppliers' enterprise systems and databases are often complicated and can influence an organization's success or failure in adopting new supply chain technologies. Finally, as mobile supply chain management involved the transmission of wireless data, they are more exposed to security and privacy risks. Organizations transmitted important supply chain information may have risks of having the information eavesdropped by hackers.

6.2.4 Organizational Factors

Firm size, senior management support, technological competence, and financial competence are the four main attributes of organizational construct used in this study. Jeyaraj et al. (2006) in their review on variables used in IT adoption studies found that top management support is consistently one of the most important predictor of IT adoption decisions. Organizations size on the other hand, have contradictory results in many IT adoption studies For example, Iacovou (1995) found that larger firms are more likely to adopt new technologies as they have far better technical and financial resources. However, Gibbs and Kramer (2004) and Huang et al. (2008) found that large organizations tend to have problem in adopting new IT systems due to their existing legacy systems which are often not compatible with newer technologies. Small organizations also tend to be more flexible when it comes to implementing new IT.

6.2.5 Environmental Factors

Environmental construct in this study is measured by competitive pressure, trading partner pressure, and expectations of market trends. Prior studies by Zhu et al. (2006) showed that when faced by competitive pressure, organizations are very likely to respond to the pressure by implementing new technologies. By implementing newer technologies, organizations hope to be able to operate more efficiently, and gain competitive advantages over their rivals. On the other hand, if majority of the organizations are adopting a new technology, those who do not follow suit may be left behind and play catch up to their competitors. This situation is very similar to when e-commerce first became popular. Many organizations that do not own a website or have e-commerce strategies look at their rivals and see that this is the industrial trend, and for them to remain competitive in the industry, they have no choice but to follow competitors in the industry to adopt e-commerce. Lastly, mobile supply chain management is often not technology that can be implemented by an independent organization. Instead, supply chain members may need to "co-adopt" the technology. This is similar to Wal-Mart's request that its top 100 suppliers need to adopt RFID (Chong et al. 2009). Therefore, it is possible that trading partner with more bargaining power will force its partners to adopt a technology.

6.2.6 Neural Network for Predicting Mobile Supply Chain Management Adoption

Neural network is a "massively parallel distributed processor made up of simple processing units, which have a natural propensity for storing experimental knowledge and making it available for use."(Haykin 2001) Similar to a human brain, neural network can acquire knowledge from its environment by learning. The knowledge acquired will be stored by the interneuron connection strengths (Haykin 2001). Neural network is capable of applying learning process to modify its synaptic weights in an orderly manner to achieve its design objective (Sexton et al. 2002).

Neural network contains nodes that are distributed in hierarchical layers. Most neural networks have an input, hidden, and output layer. Data will first go through the input layer, and the results will be generated in the output layers (Morris et al. 2004). Each input has its own synaptic weights which are transferred to the hidden layers consisting of several hidden neurons (Bakar and Tahir 2009). The values are added through applied weights before being converted into an output value.

The results will be passed to the neurons in the layers below, and provide a feed-forward path to the output layer (Sexton et al. 2002). The synaptic weights are adjusted during the repetitive training, and knowledge from the training is stored and can be used by the neural network for predictive purposes.

This research employed neural network to examine mobile supply chain adoption decisions due to several reasons. First, neural network can be both linear and nonlinear. As a result, it is an ideal method to be used for examining non-compensatory decision processes. Due to its ability to analyse nonlinear relationships, it is not necessary to assume any particular distribution for the sample population in this research (Chiang et al. 2006). Second, neural network is also found by prior studies to have better predictive performance, and this will be useful for us to understand the multi-criterion decisions involved in mobile supply chain management adoption (Chiang et al. 2006).

6.3 Methodologies

6.3.1 Data and Samples

A survey instrument was developed to test the proposed conceptual model in this research. The survey was distributed to 1,000 manufacturing firms in Malaysia via postage mail. The companies were selected from the Federation of Malaysian Manufacturers (FMM) directory which includes more than 2,000 manufacturing firms. Two weeks upon distributing the survey, phone calls and emails were made to the company to follow up with their progress in completing the survey. We requested that the survey to be completed by executives/managers in charge of supply chain management or IT practices in the organization as they will have better understanding of mobile supply chain management. Of the 1,000 surveys distributed, only 192 were returned, yielding a response rate of 19.2 %.

6.3.2 Measurement Items

The measurements items were adopted from previous studies as well as from the feedbacks from several IT organizations' managers. We adopted majority of the survey questions from previous studies.

A total of 34 questions were used to measure the independent variables. Responses to the items were made on a five-point Likert scale format ranging from 1—strongly disagree to 5—strongly agree. Mobile supply chain management was measured by three items using a five-point Likert scale format ranging from 1—strongly disagree to 5—strongly agree.

Cronbach's alpha(α) was applied to test for the reliability of the survey. The reliability coefficients (a) for each variable were as follows: Perceived costs (0.903), perceived benefits (0.696), complexity (0.621), technology integration (0.723), security perceptions (0.691), senior management support (0.714), firm size (0.694), technological competence (0.711), financial Competence (0.683), competitive pressure (0.777), expectations of market trend (0.801), trading partner pressure (0.778), and Adoption (0.737). The Cronbach's alpha (α) values ranged adhered to the suggested minimum value of 0.5 by Koh et al. (2007).

6.4 Neural Network Analyses

Multilayer perceptron training algorithm was applied to train the neural network in this research. Cross-validations were applied to prevent overfitting the model. As mentioned by Sexton et al. (2002), there is no heuristic for determining the hidden nodes in a neural network, thus a preliminary network was examined using 1–10 hidden nodes (Morris et al. 2004; Wang and Elhag 2007).

The accuracy of the model was measured using the relative percentage error over ten validations. Networks with two hidden nodes were found to be complex enough to map the datasets without incurring additional errors to the neural network model. The input layer consists of 12 predictors while the output layer consists of one output variable, which is the mobile supply chain management adoption. Table 6.2 shows the cross-validation results to determine the appropriate hidden nodes.

A tenfold cross-validation was performed where 90 % of the data (i.e., 173) was used to train the neural network, while the remaining 10 % (i.e., 19) was used to measure the prediction accuracy of the trained network. Table 6.3 shows the relative percentage error of the validations. As shown, the average cross-validated relative percentage error for the training model is 18.72 % percent while for the testing model it is 15.91 % Given that adoption decisions are complex in nature, the network model which has an accuracy of around 82 % is quite reliable in

Table 6.2 Average cross validation results with different number of hidden nodes

No. of hidden nodes	Relative percentage error
1	10.85
2	10.32
3	10.76
4	10.98
5	11.50
6	11.87
7	12.40
8	12.87
9	13.46
10	13.67

capturing the numeric relations between the predictors and the mobile supply chain management adoption.

In order to determine the ranking of the predictors in terms of their importance, sensitivity analysis was performed. Sensitivity performance was computed by averaging the importance of the input variables in predicting the output for the ten networks. The importance of an independent variable is a measure of how much the network's model-predicted value changes for different values of the independent variable. The normalized importance values were found from dividing the importance values by the largest importance value, expressed as a percentage.

Table 6.4 shows that all the predictors derived from the TOE model are relevant in all ten networks to predict mobile supply chain management adoption. Looking at the Table 6.4, the ranking of the predictors in terms of their importance are senior management support, security perceptions, technological integrations, financial and technical competence, perceived costs, complexity, trading partner pressure, competitive pressure, expectations of market trend, firm size, and perceived benefits.

Table 6.3 Full validation results of neural network model

Network	Training	Testing
1	21.23	15.45
2	18.34	15.43
3	17.45	16.23
4	22.45	17.45
5	18.87	16.32
6	16.48	15.56
7	19.34	15.35
8	18.32	16.56
9	17.87	14.32
10	16.89	16.45
Mean	18.72	15.91
Standard deviation	1.87	0.87

Table 6.4 Normalized variable importance

Predictors	Normalized importance (%)
Senior management support	92.45
Security perceptions	90.21
Technology integrations	89.33
Financial competence	88.45
Technical competence	87.21
Perceived costs	78.45
Complexity	74.32
Trading partner pressure	65.43
Competitive pressure	58.43
Expectations of market trend	55.73
Firm size	48.38
Perceived benefits	44.32

6.5 Discussions

Based on the result, senior management support is found to be the most important predictor of mobile supply chain management adoption in manufacturing firms. This means what separates those manufacturers from adopting and not adopting mobile supply chain management is the support they received from the senior management. The second most important predictor is security perceptions. This shows that given that mobile supply chain management involves transmission of data in a wireless environment, this causing some concerns for manufacturing firms in deciding to adopt mobile supply chain management.

Technology integrations are the third most important predictor in this research. Therefore, one of the hindrances for mobile supply chain management adoption is the compatibility between mobile supply chain management with the existing technology and business processes. In order to facilitate smooth integrations between mobile supply chain management with existing technologies and business processes, it would be important for the manufacturing firms to obtain the support from the top management. Both financial and technical competences are among the top half of the variables that can predict the adoption of mobile supply chain management in manufacturing firms. These two variables measure the organization's readiness. Therefore, when predicting mobile supply chain management adopters/nonadopters, manufacturing firms that are more ready in terms of financial and technical resources are more likely to adopt mobile supply chain management. In general, organizational attributes are strong predictors of mobile supply chain management adoption. The one exception of this is organization size. This could mean that our results are unlike studies which found that larger firms are more likely to adopt new technology. Smaller firms nowadays also have better technological awareness, and therefore will know the importance of mobile supply chain management. Therefore, perceived benefits are one of the worst predictor of mobile supply chain management adoption.

Environmental factors are found to have less predictive power compared to the technological and organizational factors. The results show that manufacturing firms' main decisions criteria to adopt mobile supply chain management are less likely to be influenced by trading partner pressure, competitive pressure, and expectations of market trend. However, this is not to say that these variables are not important predictors. These variables are found to be relevant in all ten neural network examinations. However, their roles in predicting mobile supply chain management adoption are less when compared to factors related to organizational and technological factors. This shows that manufacturing firms who adopt mobile supply chain management are more likely to be firms that have good resources and backed up by their senior management. Technological factors such as the perception on mobile supply chain management being secure and does not have compatibility issues will also help manufacturing firms to adopt mobile supply chain management.

6.6 Conclusion and Future Research

In an effort to determine the vital factors that contribute to the prediction of mobile supply chain management adoption, data were collected from manufacturing firms in Malaysia. Neural network was applied to the data to investigate the important predictors. All variables derived from the TOE model were found to be significant predictors of mobile supply chain management adoption. Neural network application to this research was largely successful. By using the neural network approach, practitioners and researchers are able to identify the critical variables in the mobile supply chain management adoption model.

One improvement for future studies is to incorporate more variables to predict the adoption of mobile supply chain management. Other variables from models such as Diffusion of Innovation model might be used to integrate with the TOE model. Adding additional variables might improve the predictive power of the neural network model in this research.

Acknowledgments The work described in this chapter was partially supported by the facilities provided by The Hong Kong Polytechnic University.

References

Bakar NMA, Tahir IM (2009) Applying multiple linear regression and neural network to predict bank performance. Int Bus Res 2(4):176

Chang I, Hwang HG et al (2007) Factors affecting the adoption of electronic signature: executives' perspective of hospital information department. Dec Supp Syst 44(1):350–359

Chiang WK, Zhang D et al (2006) Predicting and explaining patronage behavior toward web and traditional stores using neural networks: a comparative analysis with logistic regression. Dec Supp Syst 41(2):514–531

Chong AYL, Chan FTS (2012) Structural equation modeling for multi-stage analysis on radio frequency identification (RFID) diffusion in the health care industry. Expert Syst Appl 30(10):8645–8654

Chong AYL, Ooi KB (2008) Adoption of inter-organizational system standards in supply chains: an empirical analysis of rosettanet standards. Ind Mgmt Data Syst 108(4):529–547

Chong AYL, Ooi KB et al (2009) Factors affecting the adoption level of C-commerce: an empirical study. J Comput Inf Syst 50(2):13–22

Chou DC, Tan X et al (2004) Web technology and supply chain management. Inf Mgmt Comput Sec 12(4):338–349

Eng TY (2006) Mobile supply chain management: challenges for implementation. Technovation 26(5):682–686

Gibbs JL, Kraemer KL (2004) A cross-country investigation of the determinants of scope of E-commerce use: an institutional approach. Elec Mark 14(2):124–137

Gunasekaran A, Ngai E (2004) Information systems in supply chain integration and management. Eur J Oper Res 159(2):269–295

Haykin S (2001) Neural networks, a comprehensive foundation. Preditice Hall, Englewood Cliffs

Huang Z, Janz BD et al (2008) A comprehensive examination of Internet-EDI adoption. Inf Syst Mgmt 25(3):273–286

Iacovou CL, Benbasat I et al (1995) Electronic data interchange and small organizations: adoption and impact of technology. MIS Quart 19(4):465–485

Jeyaraj A, Rottman JW et al (2006) A review of the predictors, linkages, and biases in IT innovation adoption research. J Inf Tech 21(1):1–23

Koh SCL, Demirbag M et al (2007) The impact of supply chain management practices on performance of SMEs. Ind Mgmt Data Syst 107(1):103–124

Kuan KKY, Chau PYK (2001) A perception-based model for EDI adoption in small businesses using a technology–organization–environment framework. Inf Mgmt 38(8):507–521

Low C, Chen Y et al (2011) Understanding the determinants of cloud computing adoption. Ind Mgmt Data Syst 111(7):1006–1023

Ming Z, Xin L et al (2008) Research on mobile supply chain management based ubiquitous network. In: Proceedings of the 4th IEEE international conference on wireless communications, Networking and Mobile Computing, WiCOM

Morris SA, Greer TH et al (2004) Prediction of CASE adoption: a neural network approach. Ind Mgmt Data Syst 104(2):129–135

Ngai E, Moon KKL et al (2008) RFID research: an academic literature review (1995–2005) and future research directions. Int J Prod Econ 112(2):510–520

Sexton RS, Johnson RA et al (2002) Predicting Internet/e-commerce use. Internet Res 12(5):402–410

Sharma A, Citurs A et al (2007) Strategic and institutional perspectives in the adoption and early integration of radio frequency identification (RFID). In: Proceedings of the 40th IEEE annual Hawaii international conference on system sciences, HICSS

Tornatzky L, Fleischer M (1990) The process of technology innovation. Lexington Books, Lexington

Venkatesh V, Goyal S (2010) Expectation disconfirmation and technology adoption: polynomial modeling and response surface analysis. MIS Quart 34(2):281–303

Wang YM, Elhag T (2007) A comparison of neural network, evidential reasoning and multiple regression analysis in modelling bridge risks. Expert Syst Appl 32(2):336–348

Wang YM, Wang YS et al (2010) Understanding the determinants of RFID adoption in the manufacturing industry. Tech Soc Chan 77(5):803–815

Wu X, Subramaniam C (2011) Understanding and predicting radio frequency identification (RFID) adoption in supply chains. J Org Comput Elec Com 21(4):348–367

Xu S, Zhu K et al (2004) Global technology, local adoption: a cross-country investigation of Internet adoption by companies in the United States and China. Elec Markets 14(1):13–24

Yao Y, Palmer J et al (2007) An interorganizational perspective on the use of electronically-enabled supply chains. Dec Supp Syst 43(3):884–896

Zhu K, Kraemer KL et al (2006) The process of innovation assimilation by firms in different countries: a technology diffusion perspective on e-business. Mgmt Sci 25(10):1557–1576

Chapter 7
A Fuzzy Handling of the Multi-criteria Characteristic of Manufacturing Processes

L. Berrah and L. Foulloy

Abstract This chapter deals with the performance expression problematic in an industrial continuous improvement process. Performance expressions are the purpose of performance indicators and performance measurement systems (PMSs). We focus particularly on the elementary aspect of such an expression. The elementary performance expression is the constitutive element of the PMSs, being defined through the achievement degree of a considered objective, while other types of expressions are involved in PMSs, with regard to the multi-criteria and multilevel aspects of the objectives. The computation of the objective achievement brings together the objective declaration, the acquired measurement that reflects the reached state and the comparison of these parameters. By revisiting previous works handled in this field, we consider that elementary performance expression is modelled by a mathematical function that compares the objective to the measurement. Conventional Taylorian ratio and difference are highlighted. The qualitative or quantitative characteristic of the data, the flexibility concerning the objective declaration and the measurements errors lead us to use the fuzzy subset theory as a unified framework for expressing performance. It also leads to new approaches which are beyond comparison functions.

Keywords Industrial performance · Manufacturing process · Multi-criteria aspect · Fuzzy subset theory · Objective achievement

L. Berrah · L. Foulloy (✉)
LISTIC, University of Savoie, Annecy cedex, France
e-mail: laurent.foulloy@univ-savoie.fr

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_7,

7.1 Introduction

The major industrial control purpose is the achievement of the expected performances. In this sense, improvement processes are continuously carried out in order to define the right actions with regard to the objective achievement. Whatever the nature of the improvement process, the adopted methodology is generally based on the Deming wheel and the PDCA *Plan—Do—Check—Act* cycle (Deming 1982), and involves the following generic decision steps (PDCA 9001; Berrah et al. 2011; Monden and Ohno 2011; Womack et al. 1990):

- Identifying key areas by defining the objectives; analysing the *as-is* situation by making a diagnosis,
- Planning and implementing changes, by choosing the best improvement actions in a given context with regard to the objectives,
- Monitoring the results,
- Developing a closed-loop control system.

Making decisions implies thus a continuous check, diagnosis and action of what takes place with regard to the operational system and the target values. One major key in this context is the right information at the right moment, in order to evaluate the success of each step of the considered improvement process before going on to the next one. To be more precise, decision-making essentially needs performance expressions for the handling of both the efficacy and the efficiency of the improvement process (Berrah et al. 2008; Neely 2005; Neely et al. 1995). While the notion of efficacy is based on the objective achievement, the efficiency identifies the quantity of means used. Such requirements for the performance expression highlight, on the one hand, the multi-criteria aspect of the post-Taylorian industrial performance. The industrial performance criteria to consider hence become interrelated and concern the conventional productivity, but also quality, delay, sustainability, employee motivation, innovation, etc. On the other hand, looking to continuously improve the performances introduces specifications for the evaluation tools that include not only the performance of the *as-is* state but also the type of gap to fulfil for the reaching of the *to-be* performance (Ghalayini et al. 1997).

The problematic of the performance expression is widely considered in the literature. According to the performance indicator definition, their purpose is to give pieces of information about the objective achievement. The current measures are thus linked to the improvement actions to launch (Fortuin 1988; Bitton 1990; Berrah et al. 2000). The so-called Performance Measurement Systems (PMSs) are the instruments commonly used to reach this aim in a multi-criteria context (Globerson 1985; Bitton 1990; Kaplan and Norton 1992; Bititci 1995; Neely et al. 1995; Ghalayini et al. 1997; Clivillé et al. 2007).

This chapter deals with the "elementary" performance expression, namely the one provided by performance indicators. Conceptually, the performance expression handles data concerning the achievement of the objective. Two parameters fundamentally intervene in the expression process; on the one hand, the objective to achieve, and on the other hand the measurement of the reached state, in accordance with the objective. The elementary performance expression is obtained by the comparison of the measurement with the objective. Moreover, the semantic of the given piece of information can be different, and be either objectively related to the physical reached state or more subjectively to the satisfaction of the decision-maker with regard to the obtained results. The nature of the considered criteria as well as of the improvement to control also has its importance in the way of expressing this performance.

In (Berrah et al. 2000), a formal handling of the elementary performance expression has been proposed, based on a fuzzy modelling of the objective and the measurement. In accordance with the industrial context, the objectives can be defined with more or less flexibility. They can also be qualitatively or quantitatively declared. Beyond their possible qualitative aspect, measurements can vehicle some kind of uncertainty. The fuzzy framework allowed us to homogeneously take into account all these characteristics. Nevertheless, the idea was to consider that the comparison is made by a mathematical operator that corresponds to the nature of the objective and the measure. We propose here to consider that the specificities of the performance expression are handled first by the way the comparison is made. Hence, we choose to extend the previous propositions on a unified formal framework that involves the comparison operator and considers it as the essential parameter of the expression process. Moreover, the question about the introduction of fuzzy arguments in comparison functions is addressed once again, with regard to the semantic of the objective on the one hand and the characteristics of the measurement on the other hand.

The developed framework looks for the modelling of the elementary performance expression mechanism, namely the comparison, by a mathematical operator, of an acquired measurement with a declared objective. In this sense, Sect. 7.2 revisits the essentials around the elementary performance expression. In the spirit of applying the proposed formalism, the question of the nature of the objectives and the measurement is handled in Sect. 7.3, with regard to a match with the industrial requirements. A fuzzy handling of the objective and the measurements is thus presented, essentially based on the different semantics that the membership function concept can take. We focus in Sect. 7.4 on the way the performance can be expressed, with regard to the formats of both the objective and the measurement. Some situations, which are extracted from industrial cases, are presented in order to illustrate the different proposed formalisms.

7.2 The Performance Expression Process

A certain ambiguity remains around the performance concept, either referring to a kind of desired best level to reach or to the obtained results (Folan et al. 2007; Lebas 1995). Thus, the synergy with the “objective” and the “measurement” concepts are strong. The objective comes to quantify the performance level to reach, by planning some adequate improvement actions. The measurement identifies the reached result, once the actions have been executed. Moreover, although the objective can be declared in different ways, the performance measurement can be addressed differently. For the sake of generality, we choose to talk about "performance expression" and consider that such an expression can be associated to a reached state or a satisfaction level, with regard to this state. Hence, one can say that the performance expression reflects the achievement of an expected state, namely the objective, with regard to a physical result, which is identified to the measurement.

Intuitively, the objective concept can be considered as the performance or the satisfaction level to reach, according to a selected criterion or variable (Grabisch and Labreuche 2005; Jacquet-Lagrèze and Siskos 2001), leading to a decision-making step concerning the improvement action to launch on the considered operative system. Objectives are declared by decision-makers, in accordance with the strong and weak points of the system, *i.e.*, the set of variables or criteria to be considered for describing the given decision problem.

According to us, the human declaration of the objectives formally subscribes to the ideas that are handled by Zadeh in *Computing with Words*. The author has introduced the "precisiation" notion as a means for translating a human declaration into a formal homogeneous one, which is based on the association of a target value to a variable (Zadeh 1984, 1996, 2004). To be more precise, by using this concept, the idea is to constitute the so-called universe of discourse related to the objectives (Berrah et al. 2000), meaning the set of attributes that are required to characterise the objective. Even if the objective concept is usually identified to its target value, many parameters must be involved for its complete specification. For instance, if $v$is a variable associated with an objective, $o(v)$will represent the target value to reach, $u(v)$the unit in which the variable is expressed and $m(v)$ the measurement.

In addition, given the multi-criteria aspect of the performance (Ghalayini et al. 1997; Hon 2005; Dossi and Patelli 2008; Rezaei et al. 2011), performance expressions can be directly formulated, in the case of both a mono-objective declaration and the availability of the corresponding measurement. This is the so-called elementary expression. Performances can also be indirectly computed, with regard for instance to a multi-objective declaration, the non-availability of the measurement or a predictive estimation of the reached results. Aggregated expressions are then proposed, temporal and tendency ones also.

The kind of expression considered here is the elementary one, since only one variable is involved and the expression process is made once the measurement is acquired, at the end of the temporal horizon required to execute the associated actions. Hence, from a formal point of view, the elementary performance expression can be obtained by comparing the target value $o(v)$, to a measurement $m(v)$. The performance expression $p(v)$ describes to which extent the measurement is close to the target (Berrah et al. 2000).

Definition 1 Let v be a variable associated with an objective, the elementary performance expression is given by $p(v) = f(o(v), m(v))$ where $f : O \times M \rightarrow P$ is the comparison function, $o(v) \in O$ the target value and $m(v) \in M$ the measurement. O, M and P represent the set of values that can be taken respectively by $o(v)$, $m(v)$ *and* $p(v)$.

Note that for the sake of commensurability, we consider that $o(v)$ and $m(v)$ are comparable, *i.e.*, the values respectively associated to v with regard to the fixed objective and the way the measurement is acquired are expressed in homogeneous unities. Moreover, without loss of generality, let us assume that O and M are subsets of the positive real numbers.

The *a priori* choice of P is particularly interesting, since it vehicles the wanted semantics for $p(v)$ (Berrah et al. 2004). Two major tendencies are generally encountered in industrial practice. The first one is related to operational performance indicators. P in this case identifies the set of potential physical values that can be taken by $p(v)$.

Generally, for such indicators, P is defined on the same physical universe of discourse of $m(v)$. The second case concerns strategic or result indicators, which have to be significant and easily interpretable and comparable. P in this case identifies the set of satisfaction degrees (Figueira et al. 2005) that can be taken by $p(v)$. This satisfaction set can be numerical, e.g. $P = [0,\ 1]$, where 0 identifies a null satisfaction and 1 a total one as well as being linguistic, e.g. $P = \{very\_good, good, medium, bad, very\_bad\}$.

The choice of the comparison function constitutes a fundamental decision question. Many parameters are involved in this process, such as the nature of the improvement objective, its importance with regard to the overall strategy, the meaning handled by the comparison operation, namely some kind of precise proportion, overall profile similarity and gap to the reference. Other parameters can also be envisaged, such as the decision-maker sensibility and vision, the associate performance indicator role in the control process.

Besides, the concept of comparison is widely studied in the literature and numerous operators are proposed (Kaufmann 1976). However, from our point of view, a typology for the performance expression operators is a difficult task, except if we extend the one practiced in the Taylorian companies to the modern requirements. Indeed, in the Taylorian process where only the financial variable was considered, conventional comparison functions were the ratio and difference between the target value and the measurement. Today, comparison operators have to deal with constraints due, on the one hand to diversified values in the respective sets O, M and P, and, on the other hand, to the different natures of the involved variables.

The purpose of this work is to describe a unified framework for expressing elementary performances, by considering $p(v) = f(o(v), m(v))$. The developed idea is to propose a consolidation of the Taylorian ratio and difference operators, by focusing on their specificity. In this sense, we propose in the following section to introduce the two types of comparison function—ratio and difference—on the base of a fundamental property, which is the invariance. According to us, this concept deals with some kind of synonymy between quantitative pieces of information that are provided with regard to the performance expression. To be more precise, the invariance is highlighted when the measurement and the objective are not similar. The comparison mechanism thus handles semantics of something like the inadequacy degree. The invariance property translates the idea that two identical values $p(v_1)$ and $p(v_2)$ represents the same relation between the corresponding $(o_1(v), m_1(v))$ and $(o(v_2), m(v_2))$. Let us add that, from the measurement theory point of view (Krantz et al. 1971), the invariance idea can be associated to the "significance" concept. The significance is also associated to an operator and is deployed with regard to the different scales of values—ordinal, interval and ratio—that are used to quantify the considered variables.

7.2.1 Ratio and Difference Based Comparison Functions

A ratio formalises, from a mathematical point of view, something such as a proportion between two numbers of the same nature. Let us assume that $M \subseteq \Re$ and $O \subseteq \Re^{+*}$, where $\Re^{+*}$ is the set of strictly positive real numbers. Indeed, this hypothesis is coherent with the nature of the typical variables of industrial manufacturing systems which are generally related to positive values for the handling of both the expected state and the achieved one. Hence, the ratio $p(v) = f(o(v),\ m(v)) = \frac{m(v)}{o(v)}$ becomes a means of comparison of these pieces of information. It is obvious that if each argument of the comparison function is multiplied by the same positive real number, the ratio is unchanged. In other words, the ratio is invariant by scaling. This property conveys the idea that the obtained performance expression is overall information which is not related to the scale or the unity of the objective and the measurement. In other words, producing 90 parts for an objective of 100 parts has the same meaning in terms of performance expression as manufacturing 4,500 parts for an objective of 5,000 parts. More precisely, a service rate of 90 % due to a delay concerning 100 parts with regard to an objective of 1,000 parts has the same performance expression meaning as having a delay of 200 parts with regard to an objective of 2,000 parts. Moreover, decision-makers can ask themselves about the commensurability in this context between the 90 % performance expression related to the manufacturing objective and the one related to the service rate objective. Let us remark that it is precisely this one major problematic that is handled by the PMSs and the so-called "overall" performance expressions (Clivillé et al. 2007; Grabisch 2005).

By using a ratio, the principle of the comparison consists of comparing the objective and the measurement, not at a whole, but rather part by part. The use of the ratio comparison operator becomes relevant when the values to compare can be defined as a set that can be detailed into elements. For instance, in the industrial context, ratios were initially used to identify relationships between financial data.

Taylorian performances related to variables such as profitability, liquidity, productivity, solvency, etc., were thus expressed through them.

The difference is another operator widely used when dealing with the elements to compare as a whole. It is often the case of the delay criteria. The difference-based operator formalises, from a mathematical point of view, the gap between the entities to compare. The difference between the objective and the measurement can be written as: $p(v) = f(o(v),\ m(v)) = o(v) - m(v)$. It indicates to some extent how far from the objective the measurement is. It is also obvious that if a same real number is added to each argument of the comparison function, the resulting performance is unchanged. In other words, the difference is invariant by translation. This invariance property conveys the idea that only the difference between the objective and the measurement has a sense for the decision-maker. For example, the performance expression has the same meaning in terms of performance expression in the cases of a stock level of 55 parts, for an objective of 50 or a stock level of 5,005 parts for an objective of 5,000. Note that the same commensurability problem mentioned before can be considered in the case of the difference-based comparison.

Moreover, in particular for the difference operator, the provided results are often associated to a local use, thus allowing only the decision-makers who are directly concerned with the considered system to have an interpretation of what happened and of the quality of the result. Precisely, as mentioned before, according to the performance indicator role in the control and the decision-making process, the associated expressed performance can be defined on different P. In this sense, it seems important to us to describe the normalisation step, which is a frequent industrial practice either for the direct normalised expression of the performance or to the conversion of obtained performance expression into adapted format.

7.2.2 Normalised Ratio and Difference Operators

In the case where the performance expression represents a satisfaction degree that has to be compared or used in other information processing mechanisms, a normalisation operation is often practiced for this issue. The idea is to consider that the computed normalised expression takes its value in an interval $[p_{\text{worst}}, p_{\text{best}}]$ where p_{worst} is the worst achieved performance expression and P_{best} the best achieved one, with regard to the objective. Transforming the result provided by the comparison function into the considered normalised interval is called normalisation. In industrial practice, for the sake of simplicity, it is quite conventional to choose the interval $[0,\ 1]$.

From a formal point of view, the comparison function f previously defined is not necessarily normalised, notably depending on the choice of P and its intrinsic properties. In such cases, the normalisation can be done by means of a function $g : P \to [0, 1]$ such that $p(v) = g(f(o(v),\ m(v)))$. The properties of the function g

depend on the choice of the decision-maker and the comparison function itself. However, bounds properties are the minimal ones that are required. Indeed, at first hand, when the measurement is equal to 0, the performance expression should also be equal to 0, that is, g must be such that $g(f(x,0)) = 0$ for all $x \in O$. On the other hand, when the measurement is equal to the objective target value, the objective is fully reached and the performance expression should be equal to 1, that is, g must be such that $g(f(x,x)) = 1$ for all $x \in O$. Monotony property is also added in accordance with the two bounds.

Example 7.1 Let us assume that decision-makers consider that the performance expression $f(o(v),\ m(v)) = \frac{m(v)}{o(v)}$ represents their satisfaction degree with regard to achievement of the objective. Let us also suppose that when the measurement exceeds the objective, the satisfaction is total, implicitly meaning that the target value of the objective corresponds to the lower desired value. It is the case, for example, when sales exceed the objective. The following properties can represent this case.

- The normalised performance expression should be equal to 0 when the measurement is equal to 0: $m(v) = 0 \Rightarrow p(v) = 0$. It leads to $g(0) = 0$ and g is a non-decreasing function.
- The performance expression should be equal to 1 when the measurement is greater or equal to the objective: $m(v) \geq o(v) \Leftrightarrow \frac{m(v)}{o(v)} \geq 1 \Rightarrow p(v) = 1$.

The function g_1 such that $g_1(x) = \min(1, x)$ is a possible and quite-natural solution which satisfies the previous properties. Such a function leads to the normalised performance expression $p(v) = \min\left(1,\ \frac{m(v)}{o(v)}\right)$. In the same spirit, it would be possible to have, beyond normalisation, a smoother variation when $m(v) = 0$ and when $m(v) = o(v)$. For example, the sine-based g_2 function such that $g_2(x) = \frac{1 + \sin\frac{\pi}{2}(2\min(1,x)-1)}{2}$ provides this smooth variation. Such a function leads to the normalised performance expression $p(v) = \frac{1+\sin\frac{\pi}{2}(2\min(1,\frac{m(v)}{o(v)})-1)}{2}$.

As an illustration of the behaviour of these normalised operators, the obtained surfaces, when the objective and the measurement take their value in the subset $O = M = [0, 10] \subset \Re$, are respectively represented in Figs. 7.1 and 7.2.

Example 7.2 Let us give another example with the difference-based comparison $f(o(v),\ m(v)) = o(v) - m(v)$. The decision-maker may choose that being over performing is not fully satisfying. Such a choice implicitly means that the target value of the objective exactly corresponds to the desired value. It is the case, for example, when the performance expression is related to a stock level. Being too far from the stock level objective is sub-performing but, on the other hand, being over performing is not fully acceptable either. The following properties can represent this case.

- The performance expression should be equal to 1 when the difference is equal to 0, *i.e.*, when the measurement is equal to the objective: $m(v) = o(v) \Rightarrow p(v) = 1$. It leads to $g(0) = 1$ and g is a non-increasing function.

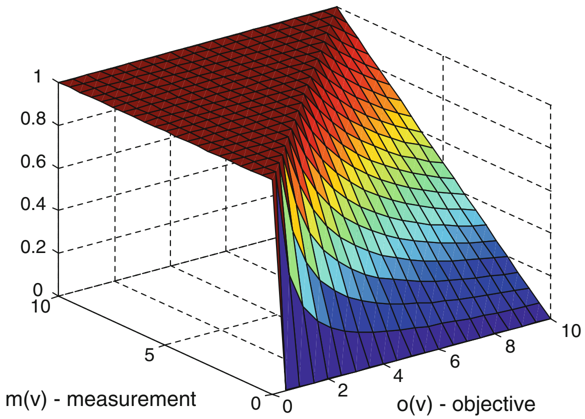


Fig. 7.1 Normalised ratio-based comparison function by g_1

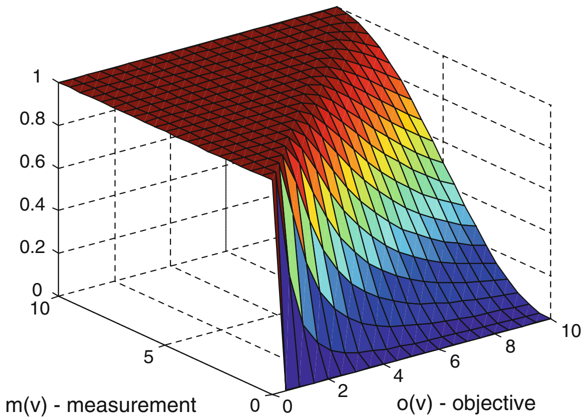


Fig. 7.2 Normalised ratio-based comparison function by g_2

- The performance expression should decrease towards 0 when the absolute value of difference is increasing. In other words, whether the measurement is over the stock level objective or not, the further from the objective, the lower the performance expression. Formally, it leads to $\lim_{|x|\to\infty} g(x) = 0$.

Among many possibilities, let us mention the sigmoid-based function g_3 such that $g_3(x) = \frac{1+e^{-\lambda c}}{1+e^{\lambda(|x|-c)}}$, where $x = c$ is the inflexion point of the sigmoid and λ is a tuning coefficient as shown in Fig. 7.3.

This function verifies the required properties and leads to the normalised performance expression $p(v) = \frac{1+e^{-\lambda c}}{1+e^{\lambda(|o(v)-m(v)|-c)}}$. Another simpler possibility is given by the function g_4 such that $g_4(x) = \frac{1}{1+\lambda|x|}$ where λ is a tuning coefficient. Such a function leads to the normalised performance expression $p(v) = \frac{1}{1+\lambda|o(v)-m(v)|}$.

As an illustration of the behaviour of these two operators, the obtained surfaces when the objective and the measurement take their value in the subset $O = M = [0, 10] \subset \Re$ are respectively represented in Figs. 7.4 and 7.5.

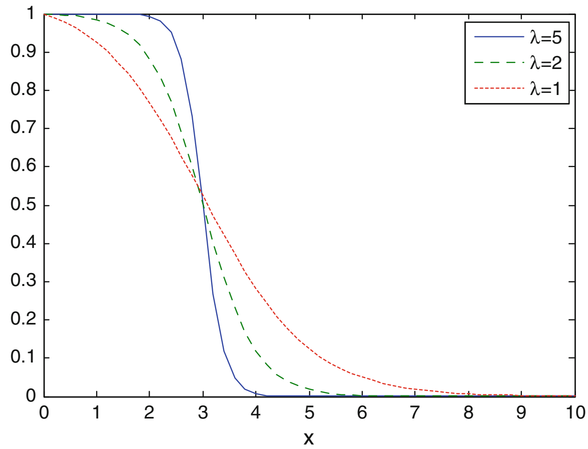


Fig. 7.3 Sigmoid function for $c = 3$.

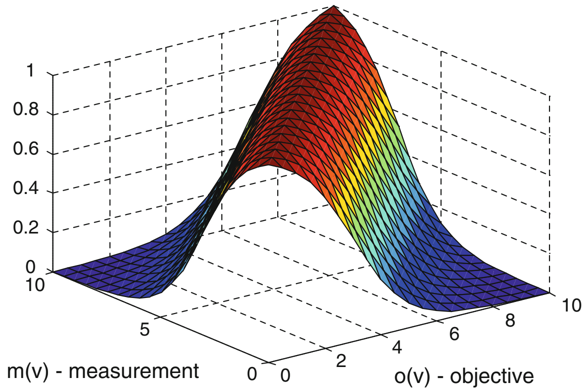


Fig. 7.4 Normalised difference-based comparisons by g_3 with $c = 3$ and $\lambda = 1$

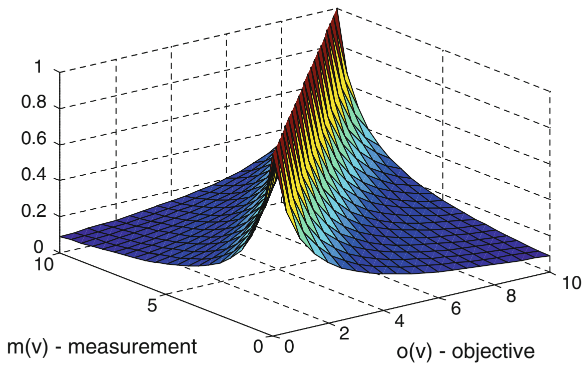


Fig. 7.5 Normalised difference-based comparison by g_4 with $\lambda = 1$

Considering once again the performance expression $p(v) = f(o(v), m(v))$, the purpose of the next section is to analyse, with regard to the industrial context, the nature of the comparison operator parameters, namely the objective $o(v)$ and the measurement $m(v)$. We thus show why they can be represented in a fuzzy way. And, in that case, the problem of the extension of the Taylorian ratio and difference operators to such data will be handled.

7.3 Fuzzy Handling of the Objective and the Measurement

In the industrial context, given the multi-criteria and multilevel aspects of the performance, associating efficacy, efficiency and effectiveness requirements to the handled improvement processes induces a compromise logic, rather than an optimisation one, between the numerous and diversified considered objectives. Moreover, the quantification of some of the involved variables can be delicate according to their qualitative or subjective way of perception or measurement. The declaration of the objectives is thus potentially characterised by some kind of nuance. Just like the objectives, by being related to the same variables, measurements can be also characterised by some nuances in their acquirement. Nuance in this case is due to the nature of the variables on the one hand and to the errors that are typical of the measurements tools on the other hand. The totality of these considerations has previously led us to use Zadeh's fuzzy subset theory for the handling of both the objective and the measurement (Berrah et al. 2000). The field of fuzzy sets has been very active since its introduction in 1965 (Zadeh 1965). This theory makes it possible to represent, from a mathematical point of view, the possibility for an element to belong partially to several sets. Hence, the concept of gradual membership to a class is introduced. This concept is particularly interesting for dealing with objectives when their declaration is made in a natural language. A few years later, Zadeh introduced the possibility theory and its link with fuzzy sets (Zadeh 1978). This theory offers an alternative way to the probability theory to deal with uncertainties (Dubois et al. 2000). If fuzzy sets are useful in the objective representation, possibility distribution is an interesting tool for dealing with measurement uncertainties. Both concepts are described in the next sections.

7.3.1 Why Can an Objective be Fuzzy?

To answer this question, one has to distinguish between the objective declaration, *i.e.*, the expression by a sentence of the objective, and the objective representation, *i.e.*, how the sentence is formally represented. The declaration step is the result of a cognitive process, which brings together different parameters that are more or less psychological (Grabisch and Labreuche 2005), dealing with bounded rationality (Simon 1982). Such a declaration is performed according to the context, on the one hand and to the behaviour of the decision-maker on the other hand; his sensitivity, his perception of what can be "good" and what cannot be "acceptable".

The representation step is, from our point of view, a kind of mapping between the reality and an adequate formalism that is able to handle in the best way this reality. This problem has been emphasised by Zadeh. By precisiation (cf. Sect. 7.1), the author has pointed out the fact that in natural language which is flexible, imprecise, uncertain, possibly missing or implicit, pieces of information are quite often used. *Precisiated Natural Language* has been introduced to deal with the description of perceptions in natural language (Zadeh 1984, 2004, 2005, 2006). Among many other examples, Zadeh has explained that "Monika is young" can be precisiated in "Age(Monika) is Young" with "Young" being a term whose meaning is defined by a membership function μ_{Young} on the set of real numbers representing the ages (Zadeh 2004). In this example, "Age" can be seen as a mapping applied to the variable "Monika".

The objective representation has been discussed in (Berrah et al. 2000) and is consolidated in the next subsections, by considering the major cases that are encountered in the industrial context. According to us, the objective declaration can be made with more or less flexibility with regard to its achievement. Moreover, this declaration can be on the numerical universe as well as the linguistic one. From a representation point of view, we propose to summarise the match between these situations in the following cases. First, numerical precise and interval-based objectives are presented with regard to a non-flexible and a flexible declaration. Then, we address another case of flexibility when such declarations are modified by adverbs or adjectives which are fuzzy or vague like almost, close to, short, long, etc. Such a modifier is called a fuzzy term from now on. Similarly to the numerical case, precise linguistic objectives are considered in a third section. Finally, the case where such linguistic objectives are modified by a fuzzy term is briefly described.

Finally, for the sake of clarity and given that no ambiguity can be possible, we choose in the following illustrations to characterise the objective by only its target value $o(v)$, without considering its universe of discourse.

7.3.1.1 Fuzzy Representation of Precise Numerical or Interval-Based Objective

Let us consider for example numerical declaration, with regard to a manufacturing objective related to a number of produced parts. The objective declarations *Produce 100 parts* and *Produce between 90 and 100* are precise numerical objectives which can be respectively written, according to our notations, $o(\text{parts}) = 100$ and $o(\text{parts}) = [90, 100]$. Their representations are respectively a singleton and an interval. Since the membership function of a fuzzy set is the extension of the characteristic function of a crisp set, their representation can be merged into the fuzzy set as shown in Fig. 7.6.

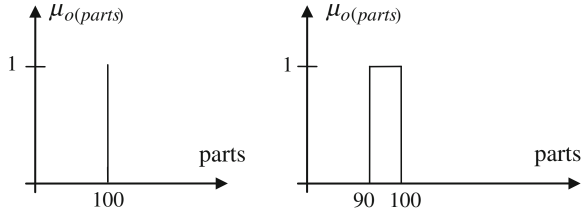


Fig. 7.6 Membership functions of precise numerical objectives

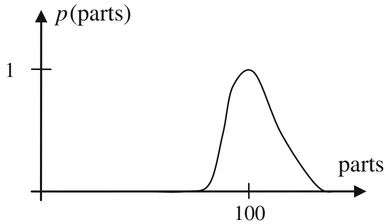


Fig. 7.7 The performance function associated with the objective *Produce about 100 parts*

7.3.1.2 Fuzzy Representation of Precise Numerical Objectives Modified by a Fuzzy Term

By considering the same manufacturing objective, let us suppose now, a slight modification of the previous precise declaration by adding the fuzzy term *about,* leading to the following declaration *Produce about 100 parts*. This example is conceptually more complex since the objective is unchanged; that is $o(\text{parts}) = 100$. A careful analysis of the semantics of the representation must be performed. It is assumed that a decision-maker is fully sincere when declaring the objective. Therefore, the objective is what should be reached without any uncertainty. Imprecision in the target value would carry an uncertainty and would mean that the decision-maker is not fully sincere. Therefore, the fuzzy expression *about 100 parts* is related to the flexibility of the satisfaction of the decision-maker. In other words, he/she would be fully satisfied if, at the end of the action plan, the number of parts is 100 but he/she would accept a lower value with a lower satisfaction. Thus, it is not because the fuzzy word *about* is used that a fuzzy number should represent the objective.

In such a case, the performance expression is no longer computed as the comparison between the objective and the measurement. Bellman and Zadeh have proposed to represent the decision-maker satisfaction by means of a membership function called the performance function (Bellman and Zadeh, 1970). Let v be a variable associated with an elementary objective, the performance expression is given by $p(v) = \mu_{o(v)}(m(v))$ where $\mu_{o(v)}$ is the membership function associated with the satisfaction of the decision-maker as shown in Fig. 7.7.

7.3.1.3 Fuzzy Representation of Precise Linguistic Objective

Linguistic declarations can be used by decision-makers in the case of an imprecise numerical quantification of the involved variable, or in the case of a lack of a numerical universe that is directly associated to the variable. The latter case is typical of a subjective evaluation or manufacturing aesthetic control quality such as, for example, the cleanliness of a ski before a silk screening operation. In such a case, there is no direct equivalent numerical universe of discourse associated to the variable and two solutions may be considered by the decision-makers. The first one consists of working directly with the linguistic term which means that the measurement itself must be also a linguistic term in order to perform the comparison at the linguistic level for the performance expression. The second solution is to consider that the linguistic objective can be related to several numerical universes of discourse on which it is possible to define an imprecise numerical quantification. In the following, for the sake of clarity, we consider that linguistic terms are always in a relation with one or several universes of discourse. This assumption is made according to industrial practice that leads decision-makers to make, in the linguistic "subjective" case, a translation into an ordinal numerical scale, such that each linguistic term is associated to a type of score.

Let us illustrate the scenario of a precise linguistic declaration case with the following example: *Produce the parts with a short delivery time*. The objective declaration is the precise linguistic term *short* which can be written, according to our notation, $o(\textit{delivery time}) = \textit{short}$. The objective needs to be "precisiated" by defining the fuzzy meaning of the term *short*. The concept of fuzzy meaning has been proposed by Zadeh to define a language as a fuzzy relation between a set of linguistics terms and a universe of discourse. The transformation between the set of linguistic terms and the related set of numbers can be formally defined by means of mapping called the fuzzy meaning and the fuzzy description (also called descriptor set by Zadeh) (Foulloy and Galichet 1995; Zadeh 1971).

The fuzzy meaning of a linguistic term takes its foundation in the relation between the linguistic terms and the universe of discourse on which they take their meaning. For example, the linguistic term *short* is linked to the universe of discourse of delivery time.

Let L be a set of linguistic terms and X a set of numbers. Let R be a fuzzy relation, *i.e.*, a fuzzy subset of the Cartesian product $L \times X$, characterised by its membership function μ_R. For any couple (l, x) in $L \times X$, the value $\mu_R(l, x)$represents the grade of membership of (l, x) to the relation R. In other words, it represents the strength of the relation between the term l and the number x. Let Z be a set, $F(Z)$ denotes the set of all fuzzy subsets of Z. The fuzzy meaning of a term l is given by the function $M : L \to F(X)$ defined by:

$$\forall l \in L,\ \forall x \in X,\ \mu_{M(l)}(x) = \mu_R(l, x) \tag{7.1}$$

In the same manner, the fuzzy description of the number x is given by the function $D : X \to F(L)$ defined by:

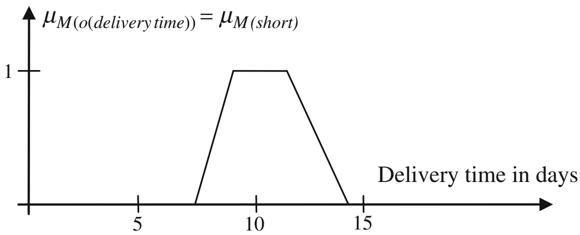


Fig. 7.8 Fuzzy meaning of the linguistic term *short*

$$\forall l \in L,\ \forall x \in X,\ \mu_{D(x)}(l) = \mu_R(l, x) \tag{7.2}$$

Since the fuzzy meaning and the fuzzy description are two ways of characterising the relation R, we have the following equality:

$$\forall l \in L,\ \forall x \in X,\ \mu_{D(x)}(l) = \mu_{M(l)}(x) \tag{7.3}$$

Thus, in order to perform the numerical-to-linguistic conversion decision-makers should be able to provide the fuzzy meaning of each term as it is represented for the term *short* in Fig. 7.8.

7.3.1.4 Fuzzy Representation of a Precise Linguistic Objective Modified by a Fuzzy Term

This case is no more than a linguistic version of the one described in Sect. 7.3.1.2. One example can be found in the declaration of the following objective, concerning the cleanliness of a ski before a silkscreen printing: *The ski should be very clean*. This objective is declared by the precise linguistic term *clean* and modified by the fuzzy term *very*. Thus, this objective can be handled thanks to a fuzzy representation. The cleanliness of a ski can be comprehended by visual analysis of the ski with respect to different parameters such as the quality of the surface, the number of small strokes, and so on, related to a numerical universe of discourse. Thus, by defining linguistic terms on these universes of discourse and by aggregating them, the fuzzy meaning of the term *clean* can be obtained. The term *very* is a linguistic modifier, called by Zadeh a linguistic "hedge" which modifies the fuzzy meaning by means of a function associated with the hedge (Zadeh 1973).

7.3.2 Why Can a Measurement be Fuzzy?

From a general point of view, it is conventional to accept errors in the measurement process. Historically, according to the used physical sensors, one way to model these errors is to quantify the uncertainties concerning the value of the measurand. The "JCGM 100:2008" guide of the Joint Committee for Guides in

Metrology defines uncertainty in measurement as a *parameter, associated with the result of a measurement, that characterizes the dispersion of the values that could reasonably be attributed to the measurand* (Evaluation of measurement data, 2008). This guide recommends basing the evaluation of the uncertainty components on probability distribution and quantifying them by variances or standard deviations. For example, the measurement can be represented by the best estimation of the measurand, generally the mean value, and by the standard deviation σ, or the standard deviation times a given number, e.g. $\pm 3\sigma$, as the characterisation of the uncertainty.

For example, let us assume that a sensor provides a reading which is 100 ± 5 and that the uncertainty is characterised by the interval $\pm 3\sigma$. Assuming a Gaussian probability distribution, it means that the value of the measurand is in the interval [95, 105] with a probability of 0.9973. Consequently, the precise value 100 can be used if uncertainties are not being taken into account. In this case, $m(v) = 100$.

However, in the case where it is necessary to take them into account, an interval can be used, e.g. [95, 105] instead of the precise value. It will lead the writing of $m(v) = [95,\ 105]$.

Even if the probability theory is predominantly used for the handling of the uncertainties, Zadeh introduced a fuzzy subset-based theory (Zadeh 1978) to represent and quantify the "possibility" for a variable to take a value. The author showed the links with fuzzy subsets by giving a semantic of possibility to the membership function, providing thus homogeneity with the objective representation. A key point of the possibility theory is thus the possibility distribution. In the case where this distribution is not directly provided by the decision-maker, one interesting idea that emerged in the 1980s and developed after the 1990s concerns the transformation of probability distribution into possibility distribution (see Dubois and Prade 1983; Civanlar and Trussel 1986; Klir and Parviz 1992; Dubois et al. 1993; Jumarie 1994, 1995) among many others, for contributions to probability-to-possibility transformations). This idea retained our interest, knowing the extension of the use of probability distributions for the handling of uncertainties in the industrial context.

Let us briefly recall the main ideas of the probability–possibility distribution. Let p be a probability distribution of a random variable X associated with the measurand and x_c be a point of the measurable space. An interval $I_{1-\alpha} = [a, b]$ around x_c is a *confidence interval* of level $1-\alpha$ if $P([a, b] = 1 - \alpha$, where P is the probability measure associated with p. One can also consider that α is the risk for the "real value" of the measurand to be outside the interval.

The probability-to-possibility proposed in (Dubois et al. 2004; Mauris et al. 2000a, 2000b), is such that the α-cut of the generated possibility distribution is the interval $I_{1-\alpha}$ which leads to:

$$\forall x \in \Re, \pi(x) = \sup(\alpha \in [0, 1]\ |x \in I_{1-\alpha}). \tag{7.4}$$

This confidence interval based construction gives the optimal probability–possibility transformation proposed by Dubois and Prade (1993). Let us remark

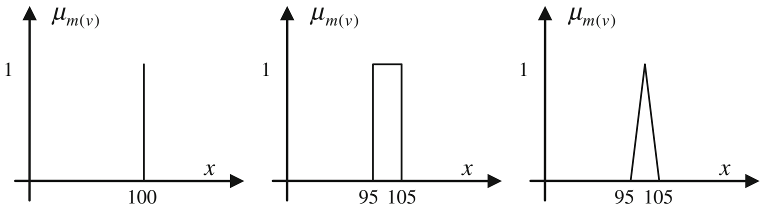


Fig. 7.9 Possibility distributions for the precise value 100, the interval [95, 105] and the optimal probability–possibility transformation of the uniform law in the bounded interval [95, 105] around the mean value 100

that the triangular possibility distribution which is widely used in applications of the fuzzy sets theory is the optimal probability–possibility transformation of the uniform law (Dubois et al. 2004).

According to us, possibility distributions provide a rich means to represent the available knowledge since it contains all the confidence intervals. A possibility distribution can be interpreted as a fuzzy subset with a semantic of uncertainty. Thus, we can write $\mu_{m(v)}(x) = \pi(x)$. It provides a unified tool since precise values and intervals can also be represented as shown in Fig. 7.9.

7.4 Computing the Performance Expression in the Fuzzy Case

The way of computing the performance expression is impacted by the nature of the objective and the measurement. This section addresses the major encountered cases, when the objective and the measurement become fuzzy.

Let us recall that we have defined the elementary performance expression by the function $f(o(v), m(v))$ whose the Taylorian ratio and difference are conventional examples. We will first extend the ratio and the difference to the fuzzy case, by using the fuzzy arithmetic operators. Besides, the questions about fuzzy objectives addressed in Sect. 7.3.2 lead to new visions of the performance expression computation, and the comparison function goes beyond a mathematical operator. In particular, a performance function can be obtained when a precise numerical objective is modified by a fuzzy term. Therefore, in a second part, we address this case when the measurement is fuzzy. A fuzzy performance is thus expressed. We have also shown that a precise linguistic objective was leading to the definition of its fuzzy meaning and the associated concept of fuzzy description (see Sect. 7.3.1.3). Thus, we consider this case when the measurement is fuzzy. The concept of fuzzy description is extended to provide an optimistic or pessimistic performance expression.

7.4.1 Elementary Performance Expression by Extending the Ratio and the Difference

We have previously assumed that both the target value and the measurement were positive real numbers. The ratio and the difference are basic arithmetic operations on real numbers, which can be extended to the fuzzy case. Indeed, the ratio and the difference become respectively the division and the subtraction between two fuzzy intervals. Arithmetic on fuzzy intervals has been widely studied since 1970 (Dubois and Prade 1983; Mizumoto and Tanaka 1976). Their definition relies on the extension of the arithmetic operation on conventional intervals for all the α-cuts of the fuzzy intervals.

Recently, Fortin et al. (2008) introduced a new vision of fuzzy intervals, which provides a new definition of the fuzzy arithmetic operators. The concept is to consider a fuzzy interval as an interval whose bounds are gradual numbers. A gradual number is defined by an assignment function from (0,1] $\rightarrow \Re$, it can be understood as a number by $\lambda \in (0,1]$ (the unit interval minus 0). Let $[x^-(\lambda),\ x^+(\lambda)]$ and $[y^-(\lambda),\ y^+(\lambda)]$ be two positive intervals, *i.e.*, $x^-(\lambda) > 0$ and $y^-(\lambda) > 0$ for all $\lambda \in (0,1]$. The four basic interval operations are:

- Addition: $[x^-(\lambda),\ x^+(\lambda)] + [y^-(\lambda),\ y^+(\lambda)] = [x^-(\lambda) + y^-(\lambda),\ x^+(\lambda) + y^+(\lambda)]$,
- Subtraction:
  $[x^-(\lambda),\ x^+(\lambda)] - [y^-(\lambda),\ y^+(\lambda)] = [x^-(\lambda) - y^+(\lambda),\ x^+(\lambda) - y^-(\lambda)]$,
- Product: $[x^-(\lambda),\ x^+(\lambda)] \cdot [y^-(\lambda),\ y^+(\lambda)] = [x^-(\lambda) \cdot y^-(\lambda),\ x^+(\lambda) \cdot y^+(\lambda)]$,
- Division: $\frac{[x^-(\lambda),\, x^+(\lambda)]}{[y^-(\lambda),\, y^+(\lambda)]} = \left[\frac{x^-(\lambda)}{y^+(\lambda)},\ \frac{x^+(\lambda)}{y^-(\lambda)}\right]$.

A fuzzy interval F, defined by its membership function μ_F is a normalised fuzzy subset of the real line such that:

- Its core is a closed interval $[f^-,\ f^+]$, actually the 1-cut of F,
- Its support is an open interval $\{x|\mu_F(x) > 0\}$,
- μ_F is non-decreasing on $(-\infty, f^-]$,
- μ_F is non-increasing on $[f^+, +\infty)$.

Let μ_{F^-} and μ_{F^+} be respectively the non-decreasing and the non-increasing part of μ_F and let these functions be injective (*i.e.*, μ_{F^-} and μ_{F^+} are respectively increasing and decreasing). Then, F is a gradual interval $[f^-(\lambda),\ f^+(\lambda)] = [\mu_{F^-}^{-1}(\lambda),\ \mu_{F^+}^{-1}(\lambda)]$.

Let us illustrate this concept with the objective and the measurement represented in Fig. 7.10.

According to the definition, the objective can be written as the gradual interval $o(v) = [90,\ 100]$ and the measurement $m(v) = [70 + 10\lambda,\ 10\lambda - 90]$.

Therefore, the extension of the ratio between the measurement and the objective to the fuzzy case is a gradual interval provided by the division which leads to $\frac{m(v)}{o(v)} = \left[\frac{70+10\lambda}{100},\ \frac{90-10\lambda}{90}\right]$. Its transformation into a fuzzy interval is represented in Fig. 7.11.

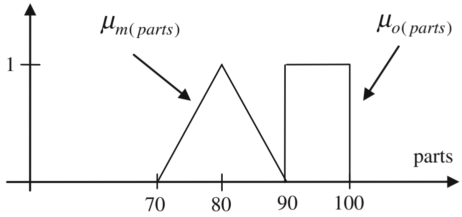


Fig. 7.10 Fuzzy measurement and objective

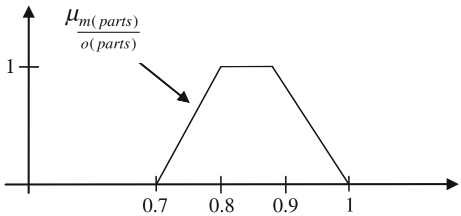


Fig. 7.11 The ratio-based performance expression

7.4.2 Elementary Performance Expression Using the Performance Function

In the case where the objective is declared with some flexibility (see Sect. 7.3.1.2), the performance expression is given by $p(v) = \mu_{o(v)}(m(v))$, when the measurement $m(v)$ is a scalar value. When the measurement is a fuzzy set, the performance expression, which is the image by the performance function, becomes itself fuzzy. In other words, the measurement uncertainties are propagated to the performance expression by the performance function. The computation relies on Zadeh's extension principle, which is recalled (Zadeh 1965).

Let $h : X \to Y$ be a function. Let A be a fuzzy subset of X whose membership function is μ_A. The image of A by the function h is a fuzzy subset of Y whose membership function μ_B is given by:

$\forall y \in Y,\ \mu_B(y) = \sup_{\{x \in X | y = h(x)\}} \mu_A(x)$ if $\{x \in X | y = h(x)\} \neq \emptyset$ and $\mu_B(y) = 0$ otherwise.

In the considered fuzzy measurement case, $\mu_{o(v)}$ is a function from O to [0, 1]. Therefore, when $m(v)$ is a scalar value then $p(v) \in [0,\ 1]$. Now if the measurement becomes fuzzy, let $\mu_{m(v)}$ be its membership function, then $p(v)$ becomes a fuzzy subset of [0, 1] whose membership function will be denoted $\mu_{p(v)}$. Zadeh's extension principle gives:

$$\forall y \in [0, 1],\ \mu_{p(v)}(y) = \sup_{\{x \in X \middle| y = \mu_{o(v)}(x)\}} \mu_{m(v)}(x) \quad \text{if} \quad \left\{x \in X \middle| y = \mu_{o(v)}(x)\right\} \neq \emptyset$$

and $\mu_{p(v)}(y) = 0$ otherwise.

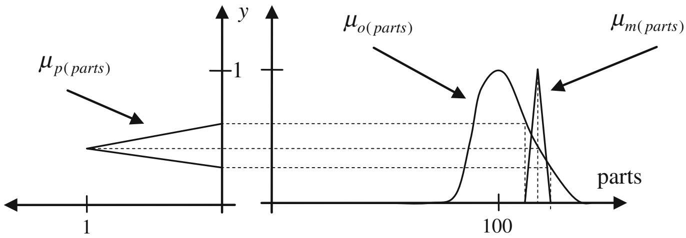


Fig. 7.12 Performance function with a fuzzy measurement

The graphical construction of the performance function is illustrated in Fig. 7.12 where, for the sake of simplicity, the resulting membership function is represented with a 90° anticlockwise rotation.

The task of the decision-maker is more complex since, as it can be observed, the performance expression is no longer a number but becomes a fuzzy number which carries the uncertainty coming from the measurement. In terms of industrial applications, even if this approach is well-founded at the formal level, the meaning of this fuzzy number is not so obvious and we think that it is better to require the measurement to be a crisp scalar.

7.4.3 Elementary Performance Expression Using Fuzzy Descriptions

When the objective is represented by the membership function which comes from the precisiation of a linguistic term, the comparison between the objective and the measurement is related to knowing to which extent the membership function associated with the measurement can be described by the linguistic term. Several comparison functions may be considered, however it is interesting to give a conceptual framework to the choice.

In order to introduce the fuzzy case, let us first examine the crisp case and assume that the meaning of the term and the measurement are crisp intervals, respectively $M(l)$ and A with $l \in L$, the set of the linguistic terms possibly characterising the objective. Let $D(x)$ be the crisp description of the number x, *i.e.*, $D(x) \subset L$ is a subset of the linguistic terms characterising the number x. Since A is an interval, each of its elements can be described by the function D leading to the family of subsets $\{D(x)\}_{x \in A}$. Therefore, it provides two bounds by taking the supremum or the infinimum of the family which are respectively called the upper and the lower description, and are defined as follows:

$$D^+(A) = \cup_{x \in A} D(x) \tag{7.5}$$

$$D^-(A) = \cap_{x \in A} D(x) \tag{7.6}$$

The upper description is an optimistic representation since it requires that only one x in A belongs to the meaning of a term to consider that the set A is described by this term. In other words, if the intersection between the set A and the meaning of a term is not empty then this term described as the set A, *i.e.*, $A \cap M(l) \neq \emptyset \Leftrightarrow l \in D^{+}(A)$.

Conversely, the lower description is a pessimistic representation since it requires that all x in A belongs to the meaning of a term to consider that the set A is described by this term. In other words, if the set A is included in the meaning of a term then this term describes the set A, *i.e.*, $A \subset M(l) \Leftrightarrow l \in D^{-}(A)$.

As explained by Foulloy and Galichet (1995), that a natural extension of this approach to fuzzy sets leads to:

$$\forall A \subset X,\ \forall l \in L,\ \mu_{D^{+}(A)}(l) = \sup_{x \in X} \min(\mu_{M(l)}(x),\ \mu_{A}(x)) \tag{7.7}$$

$$\forall A \subset X,\ \forall l \in L,\ \mu_{D^{-}(A)}(l) = \inf_{x \in X} \max(\mu_{M(l)}(x),\ 1 - \mu_{A}(x)) \tag{7.8}$$

It is interesting to note that Eqs. 7.7 and 7.8 are respectively the possibility and the necessity of the fuzzy event, characterised by the fuzzy meaning of the linguistic term l, considering the fuzzy input A as a possibility distribution. It provides a clear interpretation where the upper description gives the linguistic terms where *possibly* describe the input A while the lower description gives one where *certainly* describe it. Please remark also that when A is a crisp singleton, *i.e.*, $A = \{x\}$, we have:

$$\forall x \in X,\ \forall l \in L,\ \mu_{D^{+}(\{x\})}(l) = \mu_{D^{-}(\{x\})}(l) = \mu_{D(x)}(l) \tag{7.9}$$

Assume for example that one wants to build an optimistic comparison function to compute the performance expression, it leads to Eq. 7.10 and is illustrated in Fig. 7.13:

$$p(v) = f(o(v), m(v)) = \sup_{x \in O} \min(\mu_{M(o(v))}(x),\ \mu_{m(v)}(x)). \tag{7.10}$$

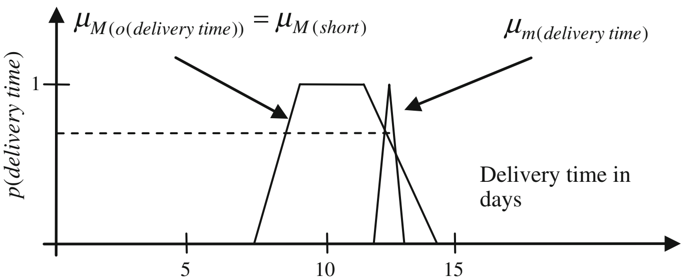


Fig. 7.13 Optimistic performance expression

7.5 Conclusion

From a general point of view, the core of the achievement of the objectives is founded in the expression of the resulted performances. By objective, we mean the target value associated to a physical or decisional variable; and by performance expression the achievement degree of this target value. The objectives are declared according to the definition of the performance concept and the induced finalities and missions of the considered system. The performances are expressed with regard to the reached states, which are described by the acquired measurements.

This chapter has dealt with the performance expression computation problematic in modern industrial companies. By revisiting previous works in this field, our proposal was to define a unified framework for such a computation, by integrating the industrial context data. In this sense, three parameters have been considered, the declared objective, the acquired measurement and the performance expression that results from the comparison of the measurement to the objective.

The conventional Taylorian performance expression operators are the ratio and the difference, leading to numerical comparison between crisp and certain values. The objective declaration is made by decision-makers and relies on expressions, which are given in natural language. These expressions can be qualitative or quantitative, precise or may convey some flexibility, some fuzziness. Just like the objectives, the measurements being associated to the same type of variables are also qualitative or quantitative. They are moreover acquired by physical sensors or human operators, leading thus to more or less uncertainty in their capture. The fuzzy sets theory proposes unified framework to address these concepts. This approach provides a means to extend the Taylorian elementary performance expression. If the most natural one is the extension to the fuzzy case of the ratio and the difference, other possibilities are offered such as fuzzy performance functions or linguistic descriptions.

Elementary performance expression is only a small part of the cases encountered in the industrial practise. More complex cases when multi-criteria information are involved provide an important field to be investigated.

References

Bellman R, Zadeh L (1970) Decision-making in a fuzzy environment. Manage Sci 17(4): B-141–B-164

Berrah L, Mauris G, Haurat A, Foulloy L (2000) Global vision and performance indicators for an industrial improvement approach. Comput Ind 43(3):211–225

Berrah L, Mauris G, Vernadat F (2004) Information aggregation in industrial performance measurement: rationales, issues and definitions. Int J Prod Res 42(20):4271–4293

Berrah L, Mauris G, Montmain J (2008) Monitoring the improvement of an overall industrial performance based on a Choquet integral aggregation. Omega 36(3):340–351

Berrah L, Montmain J, Mauris G, Clivillé V (2011) Optimising industrial performance improvement within a quantitative multi-criteria aggregation framework. Int J Data Anal Tech Strat 3(1):42–65

Bititci US (1995) Modelling of performance measurement systems in manufacturing enterprises. Int J Prod Econ 42:137–147

Bitton M (1990) Ecograi: méthode de conception et d'implantation de systèmes de mesure de performances pour organisations industrielles. PhD thesis, Thèse de doctorat en Automatique de l'Université de Bordeaux I

Civanlar M, Trussel H (1986) Constructing membership functions using statistical data. Fuzzy Sets Syst 18:1–13

Clivillé V, Berrah L, Mauris G (2007) Quantitative expression and aggregation of performance measurements based on the Macbeth multi-criteria method. Int J Prod Econ 105(1):171–189

Deming E (1982) Quality, productivity and competitive position. The MIT Press, Cambridge

Dossi A, Patelli L (2008) The decision-influencing use of performance measurement systems in relationships between headquarters and subsidiaries. Manage Acc Res 19:126–148

Dubois D, Prade H (1983) Unfair coins and necessity measures: towards a possibilistic interpretation of histogram. Fuzzy Sets Syst 10:15–20

Dubois D, Sandri S, Prade H (1993) On possibility/probability transformation. In: Lowen R, Roubens M (eds) Fuzzy logic, Kluwer Academic Press, London, pp 103–112

Dubois D, Nguyen HT, Prade H (2000) Possibility theory, probability and fuzzy sets: misunderstandings, bridges and gaps. In: Dubois D, Prade H (eds) Fundamentals of fuzzy sets, the handbooks of fuzzy sets series, Kluwer, Boston, pp 343–438

Dubois D, Foulloy L, Mauris G, Prade H (2004) Probability-possibility transformations, triangular fuzzy sets, and probabilistic inequalities. Reliab Comput 10:273–297

Evaluation of measurement data—guide to the expression of uncertainty in measurement (2008) http://www.bipm.org/utils/common/documents/jcgm/JCGM_100_2008_E.pdf

Figueira J, Greco S, Ehrgott M (2005) Multiple criteria decision analysis: state of the art surveys. Springer Publisher, New York

Folan P, Browne J, Jagdev H (2007) Performance: its meaning and content for today's business research. Comput Ind 58(7):605–620

Fortin J, Dubois D, Fargier H (2008) Gradual numbers and their application to fuzzy interval analysis. IEEE Trans Fuzzy Syst 16(2):388–402

Fortuin L (1988) Performance indicators, why, where and how? Eur J Oper Res 34:1–9

Foulloy L, Galichet S (1995) Typology of fuzzy controllers. In: Nguyen H, Sugeno M, Tong R, Yager R (eds) Theoretical aspects of fuzzy control, John Wiley and Sons, New York, pp 65–90

Free PDCA Guidance http://www.iso-9001-checklist.co.uk/iso-9001-training.htm

Ghalayini AM, Noble JS, Crowe TJ (1997) An integrated dynamic performance measurement system for improving manufacturing competitiveness. Int J Prod Econ 48(3):207–225

Globerson S (1985) Issues in developing a performance criteria system for an organisation. Int J Prod Res 23(4):639–646

Grabisch M (2005) Une approche constructive de la décision multicritère. Traitement du Signal 22(4):321–337

Grabisch M, Labreuche C (2005) Fuzzy measures and integrals in MCDA. In: Figueira J, Greco S, Ehrgott M (eds) Multiple criteria decision analysis: state of the art surveys. Springer, Heidelberg

Hon K (2005) Performance and evaluation of manufacturing systems. Cirp Annals Manu Tec 54(2):139–154

Jacquet-Lagrèze E, Siskos Y (2001) Preference disaggregation: 20 years of MCDA experience. Eur J Oper Res 130(2):233–245

Jumarie G (1994) Possibility probability transformation: a new result via information theory of deterministic functions. Kybernetics 23(5):56–59

Jumarie G (1995) Further results on possibility-probability conversion via relative information and informational invariance. Cybernet Syst 26(1):111–128

Kaplan R, Norton D (1992) The balanced scorecard: measures that drive performances. Harvard Business Review 70(1):71–79

Kaufmann A (1976) Introduction to the Theory of Fuzzy Subsets. Academic Press Inc

Klir G, Parviz B (1992) Probability-possibility transformations: a comparison. Int J Gen Syst 21:291–310

Krantz DH, Luce RD, Suppes P, Tversky A (1971) Foundations of measurement, additive and polynomial representations. Academic Press, New York

Lebas M (1995) Performance measurement and performance management. Int J Prod Econ 1–3:23–35

Mauris G, Berrah L, Foulloy L, Haurat A (2000a) Fuzzy handling of measurement errors in instrumentation. IEEE Trans Instrum Meas 49(1):89–93

Mauris G, Lasserre V, Foulloy L (2000b) Modeling of measurement data acquired from physical sensors. IEEE Trans Instrum Meas 49(6):1201–1205

Mizumoto M, Tanaka K (1976) The four operations of arithmetic on fuzzy numbers. Syst Comput Cont 7(5):73–81

Monden Y, Ohno T (2011) Toyota production system: an integrated approach to just-in-time, 4th revised edn. Productivity Press, Ohno

Neely A (2005) The evolution of performance measurement research: Developments in the last decade and a research agenda for the next. Int J Oper Prod Man 25(12):1264–1277

Neely A, Gregory M, Platts K (1995) Performance measurement system design a literature review and research agenda. Int J Prod Econ 48:23–37

Rezaei A, Çelik T, Baalousha Y (2011) Performance measurement in a quality management system. Scientia Iranica 18(3):742–752

Simon H (1982) Model of bounded rationality. MIT Press, Cambridge

Womack J, Jones D, Roos D (1990) The machine that changed the world: based on the massachusetts institute of technology 5-million dollar 5-year study on the future of the automobile. Free Press paperbacks. Rawson Associates

Zadeh L (1965) Fuzzy sets. Inform Control 8:38–353

Zadeh L (1971) Quantitative fuzzy semantics. Inform Sci 3:159–176

Zadeh L (1973) Outline of a new approach to the analysis of complex systems and decision processes. IEEE Trans Syst Man Cybern SMC 3:28–44

Zadeh L (1978) Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets Syst 1:3–28

Zadeh L (1984) Precisiation of meaning via translation into PRUF. In: Vaina L, Hintikka J (eds) Cognitive constraints on communication. Reidel, Dordrecht, pp 373–402

Zadeh L (1996) Fuzzy logic = computing with words. IEEE Trans Fuzzy Syst 2:103–111

Zadeh L (2004) A note on web intelligence, world knowledge and fuzzy logic. Data Knowl Eng 50:291–304

Zadeh L (2005) Toward a generalized theory of uncertainty (GTU)-an outline. Inform Sci 172:1–40

Zadeh L (2006) Generalized theory of uncertainty (GTU)-principal concepts and ideas. Comput Stat Data Anal 51:15–46

Chapter 8
Prioritization of Supply Chain Performance Measurement Factors by a Fuzzy Multi-criteria Approach

I. U. Sari, S. Ugurlu and C. Kahraman

Abstract Measurement of supply chain performance is an important issue to identify success, to understand processes, to figure out problems and where improvements are possible as well as provide facts for decision-making. Using classical performance measurement techniques, it may not be possible to incorporate judgments of decision makers comprehensively. Hence, we propose a fuzzy multi-criteria evaluation method for this purpose in the framework of supply chain performance measurement. Fuzzy DEMATEL is used to prioritize the performance measurement criteria of supply chain. We also present a sensitivity analysis using different linguistic scales.

Keywords Supply chain · Performance measurement · Fuzzy sets · DEMATEL method · Linguistic scale

8.1 Introduction

Globalization and the new market environment in which customer is ruling have evolved business drastically. Product life cycles have shortened significantly, agility has gained importance and outsourcing has become an option offering competitive advantage. In order to survive, collaboration among companies became inevitable. Rigid boundaries between companies have turned out to be a countercheck for performance. The ability for collaboration is encouraged through new techniques and technologies, which link a chain of companies working together as a single unit in order to satisfy customer needs. The new organization,

I. U. Sari · S. Ugurlu · C. Kahraman (✉)
Department of Industrial Engineering, Istanbul Technical University,
34367 İstanbul, Turkey
e-mail: kahramanc@itu.edu.tr

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_8,

named as supply chain (SC), is formed to achieve higher performance in the new era. In order to measure the performance of the SC, we need to combine a new perspective and novel tools with the existing performance measures. The traditional performance measures such as productivity, customer satisfaction need to be redefined in an integrated manner. On the other side, the performance factors of flexibility and risk management are considered as the ultimate purpose of a SC needed to respond to changes in the market environment.

Organizational performance measurement serves various purposes: (i) identifying success, (ii) specifying whether customer needs are met; (iii) helping to understand processes and confirming what is known or not known; (iv) identifying problems, bottlenecks, waste and where improvements are possible; (v) providing facts for decision-making; (vi) enabling and tracking improvements; and (vii) facilitating communication and cooperation (Parker 2000). In summary, a performance measurement system plays an important role in maintaining continuous improvement and decision-making.

The new organization named as SC consists of different companies with contradicting goals, technologies, and work procedures. Moreover, applications of SC aims to integrate not only various enterprises along the value chain but also various functions such as marketing, operations, sales, technology, procurement, etc., within these companies. Developing an integrated performance measurement system that would support an integrated SC development and operations is essential. The performance measures and metrics should facilitate the integration of various functional areas and also so-called extended enterprises or partnering firms along the value chain (Gunasekaran and Kobu 2007). Measuring the performance of the key functional activities of a SC is a multi-criteria decision problem. Various aspects of performance of a SC include quality, flexibility, cost (i.e., inventory turnover), customer satisfaction (i.e., responsiveness), risk (i.e., SC uncertainty), delivery (i.e., proximity to suppliers and markets).

There is a body of literature investigating SC performance measurement systems as a multi-criteria decision-making problem. Among these studies, analytical hierarchy method (AHP) and analytical network method have been used commonly. A stream of studies which employ balance scorecard is presented in order to identify the balance between external-internal focus, long-short term using the four dimensions of the scorecard. The multi-criteria nature of the problem has also been handled by operations research techniques such as mixed integer programming and data envelopment analysis. There have also been attempts to incorporate the dynamic nature of the SC in performance measurement using system dynamics and classical control theory.

In this chapter, we make use of a multi-criteria decision-making approach, which is called fuzzy decision-making trial and evaluation laboratory (DEMATEL) to prioritize the SC performance measures. We first attempt to prioritize the key performance indicators of the performance measurement system using fuzzy DEMATEL and then investigate the effect of fuzzy linguistic scale in the prioritization of the criteria.

The rest of the chapter is organized as follows: SC types and their association with performance measures are discussed in Sect. 8.2.1. Then, the scope of performance measurement in SC is reviewed in Sect. 8.2.2. In Sect. 8.3, a literature review on performance measurement systems in SC is presented. In Sect. 8.4, fuzzy set theory and linguistic variables are described. In Sect. 8.5, fuzzy DEMATEL method is reviewed. We applied fuzzy DEMATEL method for prioritization of SC performance criteria in Sect. 8.6. We then investigate the effect of the fuzzy linguistic scale on the results in Sect. 8.7. Finally, Sect. 8.7 concludes with the discussion of findings and future research.

8.2 Supply Chain Performance Measurement

In this section, we present types of SCs and performance measures as well as the scope of performance measurement in SCs.

8.2.1 Types of Supply Chains and Performance Measures

Designing a performance measurement system for a SC, performance measures and metrics should be prioritized with respect to the type of SC. In the literature, various types of SCs are identified based on the type of product manufactured. A classification of the SCs in relation to the type of products is given in Table 8.1.

Functional products are typically manufactured in high volumes so the emphasis is mainly on productivity together with quality, customer service, and cost. Demand of functional products is fairly stable. Some examples of these types of products are grocery, automobiles, etc.

However, SCs through which innovative products are manufactured, need to adapt to a volatile market. Demand is difficult to forecast. Besides, the design of

Table 8.1 Types of supply chains in the literature

Product type	Type of supply chain	Reference
Functional products	Efficient supply chain	Fisher (1997)
	Lean supply chain	Turkett (2001) Christopher and Towill (2000)
Innovative products	Quick supply chain	Fischer (1997); Huang and Uppal (2002); Selldin and Olhager (2007)
	Agile supply chain	Christopher and Towill (2000)
Functional and innovative products	Hybrid supply chain	Naylor et al. (1999); Huang and Uppal (2002)

the products changes quickly. Thus, performance of the SC depends mainly on flexibility, responsiveness and risk management. Some examples of products manufactured in quick or agile SCs are mobile phones requiring changes due to technical developments and fashion goods requiring design changes frequently.

A recent type, hybrid SC has been introduced which is a combination of a lean SC and agile SC. In hybrid SC, leanness which focuses on elimination of waste through the value stream and innovativeness are combined. For example, manufacturers of automobiles, computers, etc., need to operate in a competitive market where price is important as well as introduce innovative features in their products. Another aim for hybrid SCs is to achieve flexibility together with productivity to maintain a customer-driven approach.

Based on the focus of the SC, importance given to different performance measures and metrics may differ. For an efficient SC, productivity improvements for cost reduction and quality are vital. Cost reduction is achieved in connection to the suppliers and internal process improvements. Some of the metrics related to costs may be purchasing costs, handling costs, storage costs, supplier handling costs, etc. Similarly, quality is another important measure which is defined with many sub-dimensions such as conformance to the product specifications, performance of a product, and reliability. Some metrics related to quality are defects per million opportunities, perfect order fulfillment which calculates the error-free rate of each stage of a purchase order.

Customer satisfaction is a multidimensional performance measure for which measurements can vary greatly. For the performance measurement of SCs, delivery metrics gain importance to verify customer satisfaction. Some metrics are on time delivery, performance to promise dates or fill rate which expresses shipping performance as a percentage of the total order.

For SCs of innovative products, measurement of SC risk gains importance due to rapid change and uncertainty of markets. Risk is typically measured using the probability of an event occurring and impact of the event on the SC, and subsequently the overall business. Performance measurement systems of SCs should ensure that evaluation and redesign is made in response to market changes, including new product launches, global sourcing, new acquisitions, credit availability, the need to protect intellectual property, and the ability to maintain asset and shipment security.

Similarly, flexibility is vital for quick and agile SCs. Flexibility is needed to respond to marketplace changes to gain or maintain competitive advantage. In the literature, four types of flexibility are identified Slack (1991): (i) volume flexibility (the ability to change the output level of products produced), (ii) delivery flexibility (the ability to change planned delivery dates), (iii) mix flexibility (the ability to change the variety of products produced), and (iv) new product flexibility (the ability to introduce and produce new products). Some metrics used are SC response time and production flexibility.

8.2.2 Scope of Performance Measurement in Supply Chains

Supply chain is viewed as a new organization aiming to integrate various enterprises along the value chain. Since enterprises are building blocks of a SC, the performance of each enterprise influences the performance of its SC. Performance measurement within each enterprise is an element of the SC performance measurement system. An enterprise consists of various functions related to the SC performance. For example sourcing, production and delivery are different functions of an enterprise having different performance levels. In this perspective, the scope of the within-enterprise performance measurement may be limited to the performance of only one of the function of an enterprise named as functional performance. On the other side, within-enterprise performance measurement may be extended to cross-functional measurements along many functions of the enterprise, named as integrated performance.

From another perspective, scope of the performance measurement may be enlarged on the boundaries of an enterprise and handled together with its suppliers or customers. These types of measures are known as one-sided integrated measures and depict performance across organizational boundaries as well as measuring chain performance across supplier or customer boundaries, for example, total cost, total lead-time, and delivery speed, SC response time (Chibba 2007). However, the most complementary approach to performance measurement of SCs is depicted with the performance across organizational boundaries including links both to the suppliers and the end customers. Total chain measures are used to assess the performance of the entire SC and provide an opportunity to minimize the total cost. Stewart (1995) identified the following measures of delivery performance as total chain measures: delivery-to request date, delivery-to-commit date, and order fill lead-time.

The content of the performance measurement of SC systems is mainly related to the five phases of the SC systems: (i) plan; (ii) source; (iii) make; (iv) deliver; and (v) return. The Supply-Chain Operations Reference (SCOR) model developed by the SC Council illustrated in Fig. 8.1 summarizes the processes of a SC system.

The "plan" processes describe the planning activities associated with operating a SC, including information gathering customer requirements, resources, and balancing requirements and resources to determine planned capabilities and resource gaps. The "source" processes describe the ordering and receipt of goods and services.

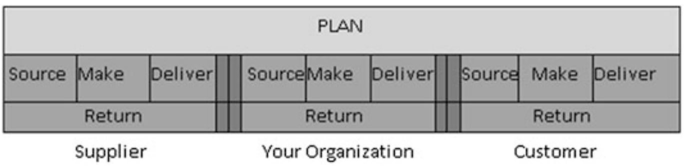


Fig. 8.1 SCOR model of the supply chain council

The "make" processes describe the activities associated with the conversion of materials or creation of the content for services which include not only production and manufacturing because but also assembly, chemical processing, maintenance, repair, overhaul, recycling, remanufacturing, and other material conversion processes. The "deliver" processes describe the activities associated with the creation, maintenance, and fulfillment of customer orders. Finally, the "return" processes describe the activities associated with the reverse flow of goods back from the customer. As is understood from the definitions of the processes of SC systems, each phase may be related to only one function of an enterprise, cross-functional in an enterprise or related to the suppliers or customers of an enterprise. Based on the level of SC performance we need to assess, the focus will be the processes of the SC in an enterprise or on the integrated performance of the whole or a part of the SC including customers or suppliers.

8.3 Literature Review

There exists a vast literature on the performance measurement systems of SCs. In the literature, seven different performance measurement systems have been proposed: function-based measurement system, dimension-based measurement system, SC operations reference model, SC balanced scorecard, hierarchical-based measurement system, interface-based measurement system, and perspective-based measurement system (Ramaa 2009). In Table 8.2, different performance measurement systems have been compared with respect to the measurement aspects and drawbacks of the system.

Performance measurement of SCs has a multidimensional nature which may be identified with the processes of the SC, management levels, performance dimensions, integration levels, or perspectives. Operations research perspective in SC performance measurement have been recently studied in the literature using data envelopment analysis, which is a nonparametric method in operations research to empirically measure productive efficiency of decision-making units. Wong and Wong (2007) developed two DEA models for the technical efficiency and the cost efficiency of internal SC performance measurement.

Talluri et al. (2006) attempted to develop a vendor evaluation model by presenting a chance-constrained data envelopment analysis approach in the presence of multiple uncertain performance measures that allow considering variability in vendor attributes. Supply chain performance is exposed to many uncertainties due to the stochastic nature of demand and supply processes. Besides, SC performance also includes many imprecise qualitative dimensions. For example, collaboration is one of the main drivers of success in SC processes. Angerhofer and Angelides (2006) developed a system dynamics model in order to reveal the constituents of a collaborative SC, key parameters they influence and pinpoint areas where the actual SC can be improved.

Table 8.2 Comparison of supply chain performance measurement systems

Performance measurement system	References	Measurement aspects	Specification of measurement	Drawback
Function based (FBMS)	Christopher (1992)	Processes of the SC	Measurement of functional processes of SC in isolation with company strategy	It does not provide the top level measures to cover the entire supply chain
Dimension based (DBMS)	Beamon (1999) Hausman (2004)	Resources, output and flexibility Service, assets and speed	Measurement based on various dimensions of performance	Dimensions should coincide with an organization's strategic goals
Supply chain operations reference model (SCOR)	Supply chain council	Reliability, responsiveness, flexibility, cost, and asset	Measurement of cross-functional processes of SC based on metrics related to processes and benchmarks	An exhaustive system requiring dedicated resources, a well-defined infrastructure, and project-based completion approach
SC balanced scorecard (SCBS)	Kaplan and Norton (1992)	Customer, internal processes, innovation and financial	Measurement based on the customers, internal business processes, learning and growth and the financial indicators	Limited to the balance scorecard dimensions
Hierarchical based (HBMS)	Gunasekaran et al. (2001)	Financial and nonfinancial metrics at strategic, tactical, and operational levels	Measurement with respect to strategic, tactical, and operational levels of management	Difficult to put measures into different levels that reduce conflicts among the supply chain partners
Interface based (IBMS)	Lambert and Pohlen (2001)	Cost, activity time, customer responsiveness, and flexibility as single or joint dimensions	Measurement of the stages of a SC which forms the total SC to optimize the total SC as well as each company	Requirement of openness and sharing of information along the chain, difficult in actual business setting
Perspective based (PBMS)	Otto and Kotzab (2003)	System dynamics, operations research, logistics, marketing, organization, strategy	Measurement of the SC in all the possible perspectives based on measures for each perspective	There can be trade-off between measures of one perspective and other perspectives

In order to handle the multidimensionality of the performance, multi-criteria decision-making methods have been employed in the literature. Bhagwat and Sharma (2007a) make use of AHP in order to combine a hierarchical performance measurement system and SC balance scorecard. They have defined the AHP model with strategic, tactical and operational levels at the upper stage and the dimensions of balance scorecard at the lower stage of AHP. Later, in 2009, Bhagwat and Sharma (2007b) propose an integrated AHP-PGP (pre-emptive goal programming) model to consider both quantitative and qualitative performance measures in optimizing the overall performance of the system. Berrah and Cliville (2007) suggests to employ a multi-criteria methodology by considering the SCOR model break-down and then an aggregation methodology, based on the Choquet integral operator and MACBETH framework. In this way, the overall performance is associated to a global objective of overall SC performance whose break-down is provided by SCOR model's elementary objectives.

Bai and Sarkis (2012) introduce an application of neighbourhood rough-set theory for the identification and selection of performance measures related to the sourcing function using the elements of SCOR model. Their model allows determining a core set of external logistics and SC performance measures to internal performance expectations and outcomes.

In this study, we propose a fuzzy multi-criteria decision-making methodology to apply a dimension based performance measurement. Our methodology first prioritizes the criteria using fuzzy DEMATEL and then we investigate the effect of linguistic variable scales. We offer a fuzzy decision-making methodology in order to include the uncertainties of SC and the imprecision of the assessment of criteria used in performance measurement system.

8.4 A Fuzzy Multi-criteria Approach

In this section, we present the basics of the fuzzy set theory and define linguistic variables. Then we briefly give the steps of fuzzy DEMATEL method.

8.4.1 Fuzzy Set Theory

The fuzzy set theory is founded by Zadeh in 1965, and he defined the fuzzy set as a class of objects with a continuum of grades of membership, which is characterized by a membership function which assigns to each object a grade of membership ranging between zero and one. A fuzzy set A in U characterized by a membership function $\mu_A(x)$ which associates with each point in U a real number in interval [0, 1], with the value of $\mu_A(x)$ at x representing "the grade of membership" of x in A (Zadeh 1965).

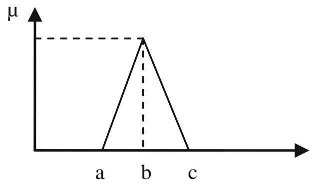


Fig. 8.2 Membership function of a TFN

A formula for membership function $\mu_A(x)$ of a triangular fuzzy number (TFN) $\tilde{x}$ which has a shape shown in Fig. 8.2, is given in Eq. (8.1), where a, b, and c denotes real numbers (Ross 1995):

$$\mu_A(x) = (l, m, r) = \begin{cases} \frac{x-a}{b-a}; & a \le x \le b \\ \frac{c-x}{c-b}; & b \le x \le c \\ 0; & \text{otherwise} \end{cases} \tag{8.1}$$

Algebraic operations for TFNs are given by (8.2)–(8.8) where all the fuzzy numbers are positive (here it is assumed to mean $a \ge 0, e \ge 0$) (Chen et al. 1992):

$$(a, b, c) + (d, e, f) \cong (a + d, b + e, c + f) \tag{8.2}$$

$$(a, b, c) - (d, e, f) \cong (a - f, b - e, c - d) \tag{8.3}$$

$$(a, b, c) \otimes (d, e, f) \cong (ad, be, cf) \tag{8.4}$$

$$(a, b, c) \div (d, e, f) \cong \left(\frac{a}{f}, \frac{b}{e}, \frac{c}{d}\right) \tag{8.5}$$

$$\lambda \otimes (a, b, c) \cong \begin{cases} (\lambda a, \lambda b, \lambda c), & \text{if} \quad \lambda \ge 0 \\ (\lambda c, \lambda b, \lambda a), & \quad \forall \lambda \in \Re \quad \lambda \le 0 \end{cases} \tag{8.6}$$

$$\lambda \div (a, b, c) \cong \begin{cases} \left(\frac{\lambda}{c}, \frac{\lambda}{b}, \frac{\lambda}{a}\right), & \text{if} \quad \lambda \ge 0 \\ \left(\frac{\lambda}{a}, \frac{\lambda}{b}, \frac{\lambda}{c}\right), & \quad \lambda \le 0 \end{cases}, \forall \lambda \in \Re \tag{8.7}$$

$$(a, b, c)^{\lambda} \cong \begin{cases} (a^{\lambda}, b^{\lambda}, c^{\lambda}), & \text{if} \quad \lambda \ge 0 \\ \left(\frac{1}{c^{\lambda}}, \frac{1}{b^{\lambda}}, \frac{1}{a^{\lambda}}\right), & \quad \lambda \le 0 \end{cases} \quad \forall \lambda \in \Re \tag{8.8}$$

8.4.2 Linguistic Variables

Linguistic variables are the variables whose values are not numbers but words or sentences in a natural or artificial language (Zimmermann 1991). Linguistic

Table 8.3 Fuzzy linguistic scale for fuzzy DEMATEL

Linguistic terms	Triangular fuzzy numbers
Very high influence (VIH)	(0.75, 1.0, 1.0)
High influence (HI)	(0.5, 0.75, 1.0)
Low influence (LI)	(0.25, 0.5, 0.75)
Very low influence (VLI)	(0, 0.25, 0.5)
No influence (NI)	(0, 0, 0.25)

variables can reflect the different levels of human language. The totality of values of a linguistic variable constitutes its term-set, which in principle could have an infinite number of elements (Zadeh 1975). In addition to the primary terms, a linguistic value may involve connectives such as and, or, either, neither, etc.; the negation not; and the hedges such as very, more or less, weakly, moderately, greatly, absolutely, etc. The hedges as well as the connectives are treated as nonlinear operators which modify the meaning of their operands in a specified fashion (Zadeh 1975).

In this study, the linguistic variable ''influence'' is used with five linguistic terms (Li 1999) as {Very high, High, Low, Very low, No} that are expressed in positive triangular fuzzy numbers (l_{ij}, m_{ij}, r_{ij}) as shown in Table 8.3.

8.4.3 Fuzzy DEMATEL Method

Decision-making trial and evaluation laboratory (DEMATEL) method, originated from the Geneva Research Centre of the Battelle Memorial Institute, is an effective method which collects group knowledge, analyzes the inter-relationships among system factors, and visualizes this structure by cause-effect relationship diagram (Gabus and Fontela 1972, 1973). The most important feature of DEMATEL in multi-criteria decision-making area is its function of building the relation and structure factors (Zhou et al. 2011). Although DEMATEL is a novel technique for evaluating problems, the relationships of systems are generally given by crisp values. The fact that human judgments about preferences are often unclear and hard to estimate by exact numerical values, necessitates fuzzy logic for handling problems characterized by vagueness and imprecision (Chang et al. 1998; Chen and Chiou 1999). Therefore, many researchers use the fuzzy DEMATEL method to extend the DEMATEL technique with fuzzy concept for making better decisions in fuzzy environments (Jeng and Tzeng 2012; Zhou et al. 2011; Chang et al. 2011; Lin and Wu 2008; Liou et al. 2008; Tseng 2009; Wu and Lee 2007).

The steps of the fuzzy DEMATEL method which is proposed by Wu and Lee (2007) are defined as follows:

Step 1: *Identifying the decision goal and forming a committee.* Decision-making is the process of defining the decision goals, gathering relevant information, generating the broadest possible range of alternatives, evaluating the alternatives for advantages and disadvantages, selecting the optimal alternative, and monitoring the results to ensure that the decision goals are achieved (Hess and Siciliano 1996; Opricovic and Tzeng 2004). Thus, the first step is to identify the decision goal. Also, it is necessary to form a committee for gathering group knowledge for problem solving.

Step 2: *Developing evaluation factors and designing the fuzzy linguistic scale*. In this step, it is necessary to establish sets of significant factors for evaluation. However, evaluation factors have the nature of causal relationships and are usually comprised of many complicated aspects. To gain a structural model dividing involved factors into cause group and effect group, the DEMATEL method must be used here. For dealing with the ambiguities of human assessments, the linguistic variable ''influence'' is used with five linguistic terms (Li 1999) as {Very high, High, Low, Very low, No} that are expressed in positive triangular fuzzy numbers (l_{ij}, m_{ij}, r_{ij}) as given in Table 8.3.

Step 3: *Acquiring and aggregating the assessments of decision makers*. To measure the relationship between evaluation factors $C = \{\, C_i |\, i = 1, 2, \ldots, n\}$, it is usually necessary to ask a group of experts to make assessments in terms of influences and directions between factors. Then, using the CFCS (Converting Fuzzy data into Crisp Scores) method, those fuzzy assessments are defuzzified and aggregated as a crisp value which is the z_{ij}. Hence, the initial direct-relation matrix $Z = \left[z_{ij}\right]_{nxn}$ can be obtained using formulas (8.9)–(8.16).

Converting fuzzy data into crisp scores method

Let $\tilde{z}^k_{ij} = \left(z^k_{lij}, z^k_{mij}, z^k_{rij}\right)$ indicate the fuzzy assessment of evaluator $k(k = 1, 2, \ldots, p)$ about the degree to which the criterion i affects the criterion j. The CFCS method includes five step algorithms described as follows:

Normalization

$$x^k_{lij} = \frac{(z^k_{lij} - \min z^k_{lij})}{\max z^k_{rij} - \min z^k_{lij}} \tag{8.9}$$

$$x^k_{mij} = \frac{(z^k_{mij} - \min z^k_{lij})}{\max z^k_{rij} - \min z^k_{lij}} \tag{8.10}$$

$$x^k_{rij} = \frac{(z^k_{rij} - \min z^k_{lij})}{\max z^k_{rij} - \min z^k_{lij}} \tag{8.11}$$

Compute left (x^k_{lsij}) and right (x^k_{rsij}) normalized values:

$$x^k_{lsij} = \frac{x^k_{mij}}{(1 + x^k_{mij} - x^k_{lij})} \tag{8.12}$$

$$x_{rsij}^{k} = \frac{x_{rij}^{k}}{(1 + x_{rij}^{k} - x_{mij}^{k})} \quad (8.13)$$

Compute total normalized crisp value:

$$x_{ij}^{k} = \frac{[x_{lsij}^{k}(1 - x_{lsij}^{k}) + (x_{rsij}^{k})^{2}]}{[1 - x_{lsij}^{k} + x_{rsij}^{k}]} \quad (8.14)$$

Compute crisp values:

$$z_{ij}^{k} = \min z_{lij}^{k} + x_{ij}^{k}\left(\max z_{rij}^{k} - \min z_{lij}^{k}\right) \quad (8.15)$$

Integrate crisp values:

$$z_{ij} = \frac{1}{p}(z_{ij}^{1} + z_{ij}^{2} + \ldots + z_{ij}^{p}) \quad (8.16)$$

Step 4: *Establishing and analyzing the structural model*. On the base of the initial direct-relation matrix $Z = \left[z_{ij}\right]_{nxn}$, the normalized direct-relation matrix $X = \left[x_{ij}\right]_{nxn}$ where $0 \le x_{ij} \le 1$, can be obtained through formula (8.17) where $i, j = 1, 2, \ldots, n$

.

$$X = \frac{1}{\max\limits_{0 \le i \le 1} \sum_{j=1}^{n} z_{ij}} Z \quad (8.17)$$

Then, the total-relation matrix T can be acquired by using formula (8.18).

$$T = X(I - X)^{-1} \quad (8.18)$$

The causal diagram can be acquired through formulas (8.19)–(8.21).

$$T = t_{ij}, \quad i, j = 1, 2, \ldots, n \quad (8.19)$$

$$D = \sum_{j=1}^{n} t_{ij} \quad (8.20)$$

$$R = \sum_{i=1}^{n} t_{ij} \quad (8.21)$$

The causal diagram is constructed with the horizontal axis (D+R) named "Prominence" and the vertical axis ($D-R$) named "Relation." The horizontal axis "Prominence" shows how much importance the factor has, whereas the vertical axis "Relation" may divide factors into cause group and effect group. Generally, when the ($D-R$) axis is plus, the factor belongs to the cause group. Otherwise, the factor belongs to the effect group if the $(D - R)$ axis is minus. Hence, causal diagrams can visualize the complicated causal relationships of factors into a visible structural model, providing valuable insight for problem solving. Further, with the help of a causal diagram, we may make proper decisions by recognizing the difference between cause and effect factors.

8.5 Performance Criteria Prioritization of Suppliers Using Fuzzy DEMATEL Method

Step 1: *Identifying the decision goal and forming a committee.*

Decision goal is defined as "prioritization of SC performance measurement criteria". The decision group consists of one general manager one manufacturing department manager and one logistics department manager.

Step 2: *Developing evaluation factors and designing the fuzzy linguistic scale*. Performance factors of SC are defined in four groups which are customer satisfaction, productivity, flexibility, and risk management due to the literature review given in Sect. 8.3.

Organizations always intend to satisfy their customers. Therefore customer satisfaction which affects all of the departments and facilities of the organizations is one of the critical factors. On time delivery (C1) and satisfying industry regulations (C2) are determined as performance criteria of customer satisfaction factor.

Productivity which is the second performance factor of suppliers is an integral part of performance. It is defined one of the most crucial area for operational and process management (Sink and Tuttle 1989; Hoehn 2003). Cost minimization (C3) and quality (C4) are determined as performance criteria of customer satisfaction factor.

Flexibility is another critical performance factor for organizations if the product type or demand could change easily. In such conditions, speed and manner of reaction (C5) and technical capability (C6) are defined as performance criteria of flexibility.

Risk management policies of suppliers have to handle the impact of the natural disasters. The sub criteria of risk management performance factor are defined as security awareness (C7), physical security (C8), and geographical location (C9). Hierarchical structure of performance factors and criteria is given in Fig. 8.3.

Step 3: *Acquiring and aggregating the assessments of decision makers.*

Influences and directions between evaluation factors are determined by the group of experts to measure the relationship between them. The influence matrices are given in Table 8.4.

Linguistic terms are expressed in positive triangular fuzzy numbers (l_{ij}, m_{ij}, r_{ij})as given in Table 8.3.

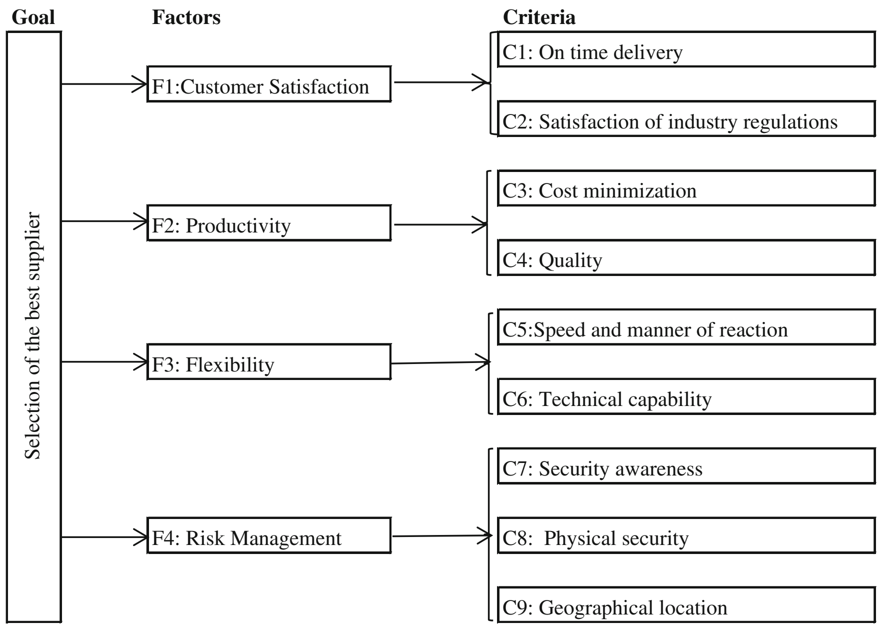


Fig. 8.3 Hierarchical structure of performance factors and criteria

Then, using the CFCS (method, fuzzy assessments are defuzzified and aggregated as a crisp value. The initial direct-relation matrix $Z = \left[z_{ij}\right]_{nxn}$ is obtained and given in Table 8.5.

Step 4: *Establishing and analyzing the structural model.*

On the base of the initial direct-relation matrix, the normalized direct-relation matrix is obtained and given in Table 8.6.

Then, the total-relation matrix is obtained and given in Table 8.7. The causal diagram is constructed with the horizontal axis $(D + R)$ and the vertical axis $(D-R)$ and given in Fig. 8.4.

The horizontal axis shows how much importance the factor has. Quality (C4) is the most important performance criteria in SC performance measurement systems whereas security awareness (C7) is the least important one. The vertical axis divides factors into cause and effect groups. We can see that technical capability (C6), geographical location (C9), and satisfaction in industry regulations (C2) are in the cause group and on time delivery (C1), cost minimization (C3), speed and manner of reaction (C5), security awareness (C7), and physical security (C8) are in the effect group. Quality (C4) is located on the vertical axis which means it has neutral effect on the other criteria. Decision makers should focus on the cause group criteria (C2, C6, and C9) and the neutral criterion (C4).

Table 8.4 The influence matrices

	C1	C2	C3	C4	C5	C6	C7	C8	C9
Expert 1_General manager									
	C1	C2	C3	C4	C5	C6	C7	C8	C9
C1	0	HI	VLI	VLI	VHI	NI	NI	NI	LI
C2	LI	0	HI	VHI	VLI	VLI	HI	HI	VLI
C3	HI	NI	0	VHI	HI	VHI	NI	LI	HI
C4	LI	HI	HI	0	LI	VHI	LI	HI	NI
C5	VHI	HI	LI	HI	0	HI	NI	LI	VLI
C6	HI	HI	VHI	VHI	VHI	0	NI	HI	NI
C7	NI	HI	VLI	VLI	VLI	VLI	0	VHI	NI
C8	NI	VLI	LI	HI	VLI	VLI	HI	0	LI
C9	VHI	NI	HI	VLI	HI	NI	NI	HI	0
Expert 2_Manager of manufacturing department									
	C1	C2	C3	C4	C5	C6	C7	C8	C9
C1	0	LI	HI	LI	HI	VLI	NI	NI	HI
C2	HI	0	HI	VHI	LI	LI	VHI	VHI	VLI
C3	HI	VLI	0	HI	VHI	VHI	NI	NI	HI
C4	HI	VHI	HI	0	LI	HI	HI	HI	VLI
C5	VHI	LI	HI	LI	0	LI	NI	NI	NI
C6	VHI	HI	HI	VHI	VHI	0	VLI	HI	NI
C7	NI	VLI	NI	LI	NI	NI	0	VHI	VLI
C8	NI	VHI	LI	HI	LI	NI	HI	0	LI
C9	HI	LI	VHI	LI	LI	LI	LI	LI	0
Expert 3_Manager of logistics department									
	C1	C2	C3	C4	C5	C6	C7	C8	C9
C1	0	HI	HI	LI	HI	NI	VLI	VLI	VLI
C2	HI	0	LI	HI	LI	LI	LI	HI	LI
C3	LI	NI	0	HI	VHI	HI	VLI	HI	LI
C4	HI	VHI	HI	0	HI	VLI	HI	VHI	VLI
C5	VHI	HI	HI	LI	0	VLI	NI	VLI	LI
C6	HI	VHI	HI	VHI	VHI	0	VLI	LI	NI
C7	NI	HI	VLI	LI	NI	NI	0	VHI	NI
C8	NI	LI	LI	LI	VLI	NI	LI	0	VLI
C9	HI	VLI	VHI	LI	VHI	VLI	VLI	LI	0

Table 8.5 Initial direct relation matrix

i	Ci1	Ci2	Ci3	Ci4	Ci5	Ci6	Ci7	Ci8	Ci9
1	0	0.66	0.58	0.41	0.82	0.11	0.11	0.11	0.50
2	0.66	0	0.66	0.89	0.41	0.41	0.74	0.82	0.34
3	0.66	0.11	0	0.82	0.89	0.89	0.11	0.42	0.66
4	0.66	0.89	0.75	0	0.59	0.66	0.66	0.82	0.18
5	0.96	0.66	0.66	0.59	0	0.50	0.04	0.26	0.26
6	0.82	0.82	0.82	0.96	0.96	0	0.18	0.66	0.04
7	0.04	0.58	0.18	0.41	0.11	0.11	0	0.96	0.11
8	0.04	0.58	0.50	0.66	0.34	0.11	0.66	0	0.41
9	0.82	0.26	0.89	0.41	0.74	0.26	0.26	0.59	0

Table 8.6 The normalized direct-relation matrix

i	Ci1	Ci2	Ci3	Ci4	Ci5	Ci6	Ci7	Ci8	Ci9
1	0	0.1	0.1	0.1	0.2	0	0.02	0.02	0.1
2	0.1	0	0.1	0.2	0.1	0.1	0.14	0.16	0.06
3	0.1	0	0	0.2	0.2	0.2	0.02	0.08	0.13
4	0.1	0.2	0.1	0	0.1	0.1	0.13	0.16	0.03
5	0.2	0.1	0.1	0.1	0	0.1	0.01	0.05	0.05
6	0.2	0.2	0.2	0.2	0.2	0	0.03	0.13	0.01
7	0	0.1	0	0.1	0	0	0	0.18	0.02
8	0	0.1	0.1	0.1	0.1	0	0.13	0	0.08
9	0.2	0	0.2	0.1	0.1	0	0.05	0.11	0

Table 8.7 The total-relation matrix

i	Ci1	Ci2	Ci3	Ci4	Ci5	Ci6	Ci7	Ci8	Ci9
1	0.34	0.43	0.45	0.43	0.48	0.26	0.22	0.33	0.28
2	0.55	0.44	0.58	0.64	0.53	0.39	0.41	0.57	0.32
3	0.57	0.46	0.48	0.62	0.62	0.47	0.29	0.48	0.37
4	0.58	0.62	0.63	0.53	0.59	0.45	0.41	0.59	0.31
5	0.56	0.49	0.53	0.52	0.41	0.37	0.25	0.4	0.28
6	0.64	0.63	0.67	0.71	0.68	0.36	0.35	0.58	0.31
7	0.23	0.33	0.28	0.33	0.25	0.18	0.17	0.4	0.16
8	0.32	0.4	0.41	0.45	0.37	0.25	0.32	0.31	0.25
9	0.54	0.43	0.57	0.5	0.54	0.33	0.28	0.46	0.24

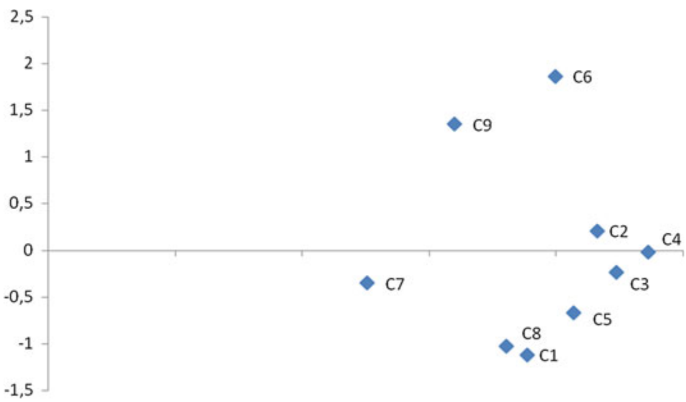


Fig. 8.4 The casual diagram of performance criteria

8.6 The Effect of the Fuzzy Linguistic Scale

There are many applications of fuzzy DEMATEL to prioritize the criteria on different decision-making problems. Mostly, the scale given in Table 8.3 is used to determine the linguistic variables. In this section, we will use another scale to

Table 8.8 Fuzzy linguistic scale for fuzzy DEMATEL

Linguistic terms	Triangular fuzzy numbers
Very high influence (VIH)	(0.7, 0.9, 1.0)
High influence (HI)	(0.5, 0.7, 0.9)
Low influence (LI)	(0.3, 0.5, 0.7)
Very low influence (VLI)	(0.1, 0.3, 0.5)
No influence (NI)	(0, 0.1, 0.3)

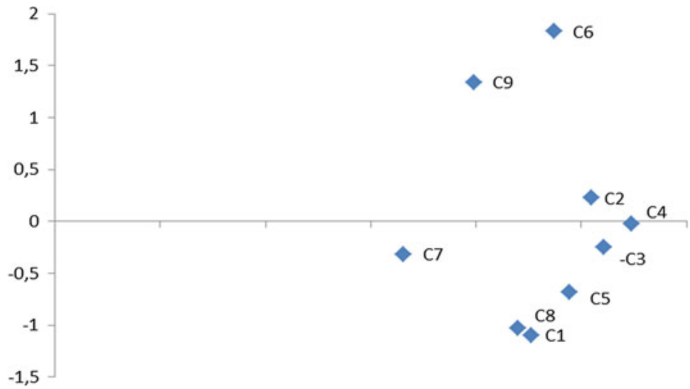


Fig. 8.5 The casual diagram of performance criteria by using Jeng and Tzeng (2012)'s scale

determine the effect of the fuzzy linguistic scale on the results. Jeng and Tzeng (2012) used the fuzzy scale which is given in Table 8.8 to determine the linguistic variables.

Jeng and Tzeng (2012)'s fuzzy linguistic scale is applied to the same influence matrix which is given in Table 8.4 and a scatter diagram is obtained and given in Fig. 8.5. We see that Figs. 8.4 and 8.5 are similar with respect to the importance rankings of the criteria as well as the grouping of cause and effect criteria. Although the places of the criteria with respect to the others do not change, only minor changes in the distances between a pair of criteria are observed. This shows that the results of fuzzy DEMATEL are robust to the selected scale of linguistic variables.

8.7 Conclusion

In this chapter, we present a review of SC performance measurement systems and offer a multi-criteria decision-making methodology, fuzzy DEMATEL in order to prioritize the performance measures of SC. Fuzzy DEMATEL enabled to collect the imprecise group judgments and analyze the inter-relationships among SC performance factors. We then visualize the factors in a cause-effect relationship diagram. We found that the most important factor of SC performance is quality. However, the structural factors which affect the other performance factors are obtained as technical capability, geographical location, and satisfaction of industry regulations. As a result, an enterprise should prioritize the improvement of technical capability, geographical location, and satisfaction of industry regulations factors since the improvement acquired in technical capability, geographical location, and satisfaction of industry regulations would also cause improvement of the factors in the effect group.

We have employed two different scales of linguistic variables in order to investigate whether the selected scale has a major effect on the results of fuzzy DEMATEL methodology. The results obtained with the use of different scales were found to be similar to each other showing that fuzzy DEMATEL is robust to the minor changes in linguistic variable scale. Our findings suggest that DEMATEL offers an effective prioritization of SC performance factors and provides a visual understanding among interrelationships of SC performance factors. For further research, the results of fuzzy DEMATEL may be used to together with analytical network process (ANP) to identify the relationships among the network structure of factors in a supplier selection problem.

References

Angerhofer BJ, Angelides MC (2006) A model and a performance measurement system for collaborative supply chains. Dec Supp Syst 42(1):283–301

Bai C, Sarkis J (2012) Supply–chain performance–measurement system management using neighbourhood rough sets. Int J Prod Res 50(9):2484–2500

Beamon BM (1999) Measuring supply chain performance. Int J Oper Prod Mgmt 19(3):275–292

Berrah L, Cliville V (2007) Towards an aggregation performance measurement system model in a supply chain context. Comput Ind 58(7):709–719

Bhagwat R, Sharma MK (2007a) Performance measurement of supply chain management using the analytical hierarchy process. Prod Plan Con 18(8):666–680

Bhagwat R, Sharma MK (2007b) Performance measurement of supply chain management: A balanced scorecard approach. Comput Ind Eng 53(1):43–62

Chang B, Chang CW, Wu CH (2011) Fuzzy DEMATEL method for developing supplier selection criteria. Expert Syst Appl 38(3):1850–1858

Chang YH, Yeh CH, Cheng JH (1998) Decision support for bus operations under uncertainty: A fuzzy expert system approach. Omega 26(3):367–380

Chen LH, Chiou TW (1999) A fuzzy credit–rating approach for commercial loans: A Taiwan case. Omega 27(4):407–419

Chen SJ, Hwang CL, Hwang FP (1992) Fuzzy multiple attribute decision making: methods and applications. Lecture notes in economics and mathematical systems, Springer–Verlag, Berlin

Chibba A (2007) Measuring supply chain performance measures prioritizing performance measures. Licentiate thesis, Luleå University of Technology, Sweden

Christopher M (1992) Logistics and supply chain management. Pitman Publishing, London

Christopher M, Towill D (2000) Supply chain migration from lean and functional to agile and customized. Supply Chain Mgmt: Int J 5(4):206–213

Fisher ML (1997) What is the right supply chain for your product?. Harvard Bus Rev March–April Reprint number: 97205, pp 105–116
Gabus A, Fontela E (1972) World problems, an invitation to further thought within the framework of DEMATEL. Battelle Geneva Research Centre, Switzerland
Gabus A, Fontela E (1973) Perceptions of the world problematic: communication procedure, communicating with those bearing collective responsibility (DEMATEL report no. 1). Battelle Geneva Research Centre, Switzerland
Gunasekaran A, Kobu, B (2007) Performance measures and metrics in logistics and supply chain management: a review of recent literature for research and applications. Int J Prod Res 45(12): 2819-2840
Gunasekaran A, Patel C, Tittiroglu E (2001) Performance measures and metrics in a supply chain environment. Int J Oper Prod Mgmt 2(1–2):71–87
Hausman WH (2004) Supply chain performance metrics. In: Harrison TP, Lee HL and Neale JJ (eds) The practice of supply chain management: where theory and application converge. Springer Science and Business, New York, pp 61–73
Hess P, Siciliano J (1996) Management: responsibility for performance. McGraw–Hill, New York
Hoehn W (2003) Managing organizational performance: linking the balanced scored to a process improvement technique. In: Proceedings of the 4th annual international symposium in industrial engineering on the performance–based management. Kasetsart University, Bangkok, pp 1–12
Huang SH, Uppal M (2002) A product driven approach to manufacturing supply chain selection. Supply Chain Mgmt: Int J 7(4):189–199
Jeng DJF, Tzeng GH (2012) Social influence on the use of clinical decision support systems: revisiting the unified theory of acceptance and use of technology by the fuzzy DEMATEL technique. Comput Ind Eng 62(3):819–828
Kaplan RS, Norton D (1992) The balanced scorecard –measures that drive performance. Harvard Bus Rev 70:71–9
Lambert DM, Pohlen TL (2001) Supply chain metrics. Int J Logist Mgmt 12(1):1–19
Li RJ (1999) Fuzzy method in group decision making. Comput Math Appl 38(1):91–101
Lin CJ, Wu WW (2008) A causal analytical method for group decision–making under fuzzy environment. Expert Syst Appl 34(1):205–213
Liou JJH, Yen L, Tzeng GH (2008) Building an effective safety management system for airlines. J Air Transp Mgmt 14(1):20–26
Naylor JB, Naim MM, Berry D (1999) Leagility: integrating the lean and agile manufacturing paradigms in the total supply chain. Int J Prod Eco 62(1–2):107–118
Opricovic S, Tzeng GH (2004) Compromise solution by MCDM methods: a comparative analysis of VIKOR and TOPSIS. Euro J Oper Res 156(2):445–455
Otto A, Kotzab A (2003) Does supply chain management really pay? Six perspectives to measure the performance of managing a supply chain. Euro J Oper Res 144(2):306–320
Parker C (2000) Performance measurement. Work Study 49:63–66
Ramaa A, Rangaswamy TM, Subramanya KN (2009) A review of literature on performance measurement of supply chain network. In Proceedings of the 2nd international conference on emerging trends in engineering and technology, ICETET–09
Ross TJ (1995) Fuzzy logic with engineering applications. McGraw-Hill, New York
Selldin E, Olhager J (2007) Linking products with supply chains testing Fischer's model. Supply Chain Mgmt: Int J 12(1):42–51
Sink D, Tuttle T (1989) Planning and measurement in your organization of the future. IE Press, Norcross
Slack N (1991) The manufacturing advantage. Mercury Books, London
Stewart G (1995) Supply chain performance benchmarking study reveals keys to supply chain excellence. Logist Inf Mgmt 8(2):38–44
Talluri S, Narasimhan R, Nair A (2006) Vendor performance with supply risk: a chance–constrained DEA approach. Int J Prod Eco 100(2):212–222

Tseng ML (2009) A causal and effect decision making model of service quality expectation using grey–fuzzy DEMATEL approach. Expert Syst Appl 36(4):7738–7748

Turkett RL (2001) Lean manufacturing implementation–lean supply chain. Notes for the course of “IOE, Manufacturing Strategies”, www.engin.umich.edu/class

Wong WP, Wong KY (2007) Supply chain performance measurement system using DEA modeling. Ind Mgmt Data Syst 107(3):361–381

Wu WW, Lee YT (2007) Developing global managers’ competencies using the fuzzy DEMATEL method. Expert Syst Appl 32(2):499–507

Zadeh LA (1965) Fuzzy Sets Inf Con 8:338–353

Zadeh LA (1975) The concept of a linguistic variable and its application to approximate reasoning – I. Inf Sci 8:199–249

Zhou Q, Huang W, Zhang Y (2011) Identifying critical success factors in emergency management using a fuzzy DEMATEL method. Safety Sci 49(2):243–252

Zimmermann HJ (1991) Fuzzy set theory and its application, 2nd edn. Kluwer, Dordrecht

Chapter 9
Route Selection and Consolidation in International Intermodal Freight Transportation

R. A. Kumar, P. Mohapatra, W. K. Yew, L. Benyoucef and M. K. Tiwari

Abstract This chapter focuses on selecting the route in international intermodal freight transportation network considering the following characteristics, first and foremost multi-objective: minimization of travel time and travel cost, later schedules and delivery times of every service provider in each pair of location, and lastly variable cost must be included in every location. The study aims to formulate the problem into mixed integer linear programming (MILP) model and develop an algorithm which encompassing all the above essential characteristics. It is NP-hard problem; it follows the proposed algorithm (nested partitions method) that is heuristic and multi-attribute decision-making (MADM) method. An illustrative experiment is considered and our proposed algorithm is applied to obtain an effective and efficient solution.

Keywords Intermodal · Mixed integer linear programming · Nested partition method · multi-attribute decision making

9.1 Context and Motivation

In the recent years, the importance of intermodal freight transportation is increased in global business transaction, hoping it substantially reduces logistics costs. Intermodal freight transportation can be defined as movement of cargos from

R. A. Kumar · P. Mohapatra · M. K. Tiwari (✉)
Department of Industrial Engineering and Management, Indian Institute of Technology, Kharagpur, India
e-mail: mkt09@hotmail.com

W. K. Yew
Department of Manufacturing and Industrial Engineering, Universitiy Teknologi, Skudai, Malaysia

L. Benyoucef
Aix-Marseille University, LSIS UMR 7296 13397 Marseille Cedex 20, France

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_9,

origin to destination using two or more transportation modes such as air, sea, road, and rail. Though intermodal transportation reduces the logistics cost, there are some operational challenges such as, how an international intermodal carrier selects best routes for shipments through the international intermodal network, which is explained in upcoming sections.

Our concentration is only on ocean and air transportation modes. Intermodal routing problem is complicated by following important characteristics; first it is important to include multiple objectives such as minimization of travel time and travel cost because customers may have different concerns, second transportation mode schedules and delivery times of every service provider in each pair of location must be included in the modeling of routing, and last the demand delivery times could be treated as time window constraints. Each location may have its own security cost, handling cost, etc., it has to be considered as variable cost.

The remainder of this chapter is organized as follows. Section 9.2 reviews the state-of-the-art related to intermodal routing methods. Section 9.3 presents the mixed integer linear programming (MILP) formulation of the problem under consideration. Section 9.4 describes the evaluation techniques. Section 9.5 shows the implementation of the techniques and methods. Section 9.6 concludes the chapter.

9.2 Literature Review

Recently, there are several researches to develop intermodal routing methods and consolidation. So far, the research articles which focus on intermodal routing (Crainic and Rousseau 1986; Min 1991; Barnhart and Ratliff 1993; Boardman et al. 1997; Bookbinder and Fox 1998; Southworth and Peterson 2000; Chang 2008; Shi and Olafsson 2008; Tadashi et al. 2009; Çakır 2009; Leung et al. 2009; Rafay and Charles 2010; Zhang et al. 2010; Kang et al. 2010; Hu 2011; Yang et al. 2011 and Cho et al. 2012).

Crainic and Rousseau (1986) examined the multimode, multicommodity freight transportation problem which occurs when the same authority controls and plans both the supply of transportation services and the routing of freight. The problem is solved by means of an algorithm based on decomposition and column generation principles.

Boardman et al. (1997) developed a decision support system, to find out the least cost routes by *K* shortest path double sweep method. They considered only transport cost of all the modes and they obtained a least cost routes by combining the modes and obtained a reduced cost route.

Chang (2008) developed a mathematical model to encompass the three essential characteristics of international intermodal routing network (multiple objectives, schedule transportation modes and demand delivery times and transportation economics). He proposed an algorithm based on relaxation and decomposition technique, which can effectively and efficiently solve the problem with time window and concave costs. He used Lagrangian multiplier to relax the problem, after relaxing he decomposed the problem into two, further minimization of time including cost, later he still decomposed it to sub problem, then he used branch and bound technique to solve those sub problems, finally he obtained an optimal solution with both minimum cost and minimum time.

Tadashi et al. (2009) designed a multimodal freight transportation network within the framework of bi-level programming and used heuristic approach based on genetic local search as a solution technique. Leung et al. (2009) addresses the problem of determining the optimal integrations and consolidations of air cargo shipments.

Rafay and Charles (2010) developed a strategic planning for an interregional, hub-based, intermodal logistics network operated by a logistics service provider. Real-time study on US freight flows and solved it using tabu search meta-heuristic model.

Kang et al. (2010) used PSO-ACO double-layer optimization algorithm. The PSO is performed at the master level to select nodes sequence, while the ACO is carried out at the slave level to look for optimal transportation mode and path combination. Zhang et al. (2010) proposed a novel method, which combines artificial immune system (AIS), chaos operator, and particle swarm optimization (PSO).

Hu (2011) proposed an integer linear programming model to build the path selection for container supply chain in the context of emergency relief. The simulation study shows the promising effects of the model. Yang et al. (2011), they proposed a model based on the principles of goal programming.

Cho et al. (2012) suggested a dynamic programming algorithm to minimize cost and time simultaneously and they used Label setting algorithm and MADM model to arrive at an optimal solution.

Most of the existing works varied with different characteristic based on the objectives (Crainic and Rousseau 1986; Boardman et al. 1997; Chang 2008; Çakır 2009; Rafay and Charles 2010; Rahim and Farhad 2011; Hu 2011; Cho et al. 2012). To the most extent, Chang (2008) and Cho et al. (2012) had covered some essential characteristics which this study is interested on. Chang formulated the problem into two objective functions later to solve; he developed an algorithm based on relaxation and decomposition techniques.

The objective of this research work is how to select the best routes in an international intermodal routes network problem. For this problem, we also need to consider the service providers available in each and every pair of nodes and the problem becomes more complex by including customers concern to select their best route. In this chapter, we develop an algorithm encompassing all the above essential characteristics (multiple objectives, service provider's time schedule, travel time and delivery time, commodity allowed, capacity of modes, and customers' choice). We propose two different methods to obtain the optimal route; it follows the proposed algorithm 'NP' (nested partition method) developed by Olafsson et al. (2003) and 'MADM' (multi-attribute decision-making) method.

The NP method is applied to substructure original complex problem into simple sub problem, through which we can obtain a set of feasible routes in the route network. Later, the set of feasible routes goes through the MADM evaluation techniques in here we can implement the customer's choice and obtain the best available route in the international intermodal routing network.

9.3 Problem Description and Formulation

Considering a directed transportation network, $G\ (N, E)$ where set of nodes be N, and set of links be E. The time window of node $i \in N$ is $[l_i, u_i]$, where the earliest arrival time be l_i and the latest acceptable arrival time be u_i, $l_i \leq u_i$ and $u_i - l_i < \infty$. Let the arrival time at node i of service provider s be denoted asA_i^s, the link travel time from node i to node j by service provider t_{ij}^s, and the service time at node i for service provider s be S_i^s. The departure time from node i by service provider $s$can then be represented by $A_i^s + S_i^s$. However, some service providers may have scheduled departure time in few nodes; such time can be represented as $D_i^s$from node i and service provider$s$. If the scheduled departure time for service provider s from node i does not exist then D_i^s is equated to zero. Consequently, the departure time for any service provider from node $i$can be represented by $\max\{D_i^s, A_i^s + S_i^s\}$.

For each service provider s in the link (i,j), the cost is assumed to be a continuous non-convex piecewise linear function of the total flow along the link. A set R of capacity ranges is also given for each link (i,j). Each range r is associated with two types of costs in every service provider, a variable cost $C_{ij}^{sr}$and a fixed cost F_{ij}^{sr} which is incurred only when the total flow on link $(i,j)$is within the range r and service provider s. Consider a set V of communicating node pairs. Every pair $c \in V$ is treated as a distinct commodity, and is associated with an origin node $O(c)$ and a destination node $D(c)$. Assume that d_c units of flow must be sent from $O(c)$ to $D(c)$.

The following notations are used in the model.

N	The set of nodes in the network
E	The set of links in the network
S	The set of available service providers in every link
R	The set of available capacity ranges in every service provider
d_c	The traffic requirement for commodity c
$O(c)$	The source node for communicating node pair c
$D(c)$	The destination node for communicating node pair c
C_{ij}^{sr}	The variable cost for link (i, j), service provider s with range r
F_{ij}^{sr}	The fixed cost for link (i, j), service provider s with range r

$$Y_{ij}^{sr} \begin{cases} 1 & \text{if service provider } s \text{ is selected with range } r \text{ on the link } (i,j). \\ 0 & \text{otherwise} \end{cases}$$

The objective function of the problem can now be formulated as follows:

$$P_1 = \sum_{ij} \sum_{s} \sum_{r} \sum_{m} F_{ij}^{sr} \cdot Y_{ij}^{sr} \cdot M_{ij}^{m} + \sum_{\mathrm{ij}} \sum_{\mathrm{s}} \sum_{\mathrm{r}} \sum_{c} \sum_{m} C_{ij}^{sr} \cdot X_{ij}^{cr} \cdot M_{ij}^{m} \tag{9.1}$$

$$\mathrm{P}_2 = \sum_{ij} \sum_{s} \sum_{c} \sum_{m} [\max(D_j^s,\ \mathrm{A}_{\mathrm{j}}^{\mathrm{s}} + \mathrm{S}_{\mathrm{j}}^{\mathrm{s}}) - \max(\mathrm{D}_{\mathrm{i}}^{\mathrm{s}},\ A_i^s + S_i^s)] \cdot M_{ij}^{m} \cdot W_{ij}^{c} \tag{9.2}$$

Subject to

$$\sum_j W_{ij}^c - \sum_j W_{ji}^c = \begin{cases} d_c & \text{if } i = O(c) \\ -d_c & \text{if } i = D(c) \\ 0 & \text{otherwise} \end{cases} \tag{9.3}$$

$$W_{ij}^c \geq \sum_r X_{ij}^{cr} \quad \text{for all } (i,\ j) \in E,\ c \in V \tag{9.4}$$

$$X_{ij}^{cr} \leq d_c Y_{ij}^{sr} \text{ for all } (i,j) \in E, c \in V,\ s \in S,\ r \in R \tag{9.5}$$

$$X_{ij}^{cr} \leq d_c Y_{ij}^{sr} \quad \text{for all } (i,j) \in E, c \in V, s \in S, r \in R \qquad \sum_s Y_{ij}^{sr} \leq 1 \tag{9.6}$$

$$\sum_m M_{ij}^m \geq 1 \tag{9.7}$$

$$W_{ij}[\max(D_i^s, A_i^s + S_i^s) + t_{ij}^s - A_j^s] = 0 \tag{9.8}$$

$$l_j \leq A_j^s \leq u_j \tag{9.9}$$

$$A_{j-1}^s < A_j^s \tag{9.10}$$

$$X_{ij}^{cr} \text{ non--negative integer for all } (i,j) \in E,\ c \in V,\ r \in R \tag{9.11}$$

$$W_{ij}^c \text{ non} - \text{negative integer for all } (i,j) \in E\ c \in V \tag{9.12}$$

$$Y_{ij}^{sr} \in \{0, 1\} \quad \text{for all } (i,j) \in E,\ s \in S,\ r \in R \tag{9.13}$$

$$M_{ij}^m \in \{1, 2\} \quad \text{for all } (i,j) \in E,\ m \in \{1, 2\} \tag{9.14}$$

In the above formulation, the first objective P_1 minimizes the total flow cost. The second objective P_2 minimizes the total travel time. Constraint set (9.3) is the flow conservation equations. Constraints in set (9.4) ensure that the flow of commodity c on link (i,j) does not exceed the total flow of the same commodity in all possible ranges of link (i,j). Constraint set (9.5) forces to choose service provider s in the range r on link (i,j) if commodity c has flow through it. Constraint set (9.6) ensures at most one service provider is selected for each link. Constraint set (9.7) ensures that mode m to be a mandatory and it must be one or more than one, which is related with constraint (9.14). Constraint set (9.8) ensures the compatibility requirements between flow and time variables. Constraint set (9.9) specifies the time window associated with each node. Constraint set (9.10) specifies that the arrival time of predecessor node is always less than the arrival time of current node for a service provider s. Constraint sets (9.11)–(9.14) are non-negativity constraints.

In the Chang's study, the two objective functions are combined into a single objective function using weighted method and he generalized it into a cost function, later solved the same using Lagrangian relaxation and decomposition techniques. Since this study's main objective is to get the best route on customers concern. Therefore, it follows two evaluation techniques such as (1) nested partition method and (2) multi-attribute decision-making evaluation.

9.4 Evaluation Techniques

9.4.1 Nested Partition Method

The *nested partitions method* (NP), a relatively new optimization method that has been found to be very effective solving discrete optimization problems. Such discrete problems are common in many practical applications, and the NP method is thus useful in diverse application areas. It can be applied to both operational and planning problems and has been demonstrated to effectively solve complex problems in both manufacturing and service industries.

NP method has been successful in solving complex problems in planning and scheduling, logistics and transportation, supply chain design, data mining, and health care. The NP method is best viewed as a metaheuristic framework and it has similarities to branching methods in that like branch-and-bound it creates partitions of the feasible region. However, it also has some unique features that make it well suited for very difficult large-scale optimization problems given by Shi et al. (1999); Shi and Olafsson (2000), (2008), and Sigurdur (2003).

Nested Partitions Framework

Step 1 Partitioning

This step partitions the current most promising region into several sub regions and aggregates the remaining regions into the surrounding region. With an appropriate partitioning scheme, Partition the most promising region $\sigma(k)$ into M subregions $\sigma_1(k), \ldots, \sigma_M(k)$, and aggregate the complimentary region $X/\sigma(k)$ into one region $\sigma_{M+1}(k)$.

Step 2 Random Sampling

Randomly generate N_j sample solutions from each of the regions $\sigma_j(\mathrm{k}),\ j = 1, 2, \ldots, M + 1$:

$$X_1^j, X_2^j, \ldots, X_{N_j}^j,\ j = 1, 2, \ldots, M + 1$$

Calculate the corresponding performance values:

$$f(x_1^j), f(x_2^j), \ldots, f(x_{N_j}^j),\ j = 1, 2, \ldots, M + 1$$

Step 3 Calculation of the Promising Index

For each region, we calculate the promising index to determine the most promising region. For each region σ_j, $j = 1, 2, \ldots, M + 1$, calculate the promising index as the best performance value within the region:

$$I(\sigma_j) = \min_{i=1,2,\ldots,N_j} f(x_i^j),\ j = 1, 2, \ldots, M + 1$$

Step 4 Backtracking

The new most promising region is either a child of the current most promising region or the surrounding region. If more than one region is equally promising, ties are broken arbitrarily. When the new most promising region is the surrounding region, backtracking is performed. The algorithm can be devised to backtrack to either the root node or any other node along the path leading to the current promising region.

Evaluating the partial solution by branching process on the overall solution space, each time a partial solution is sampled and evaluated, the overall solution space is actually branched into two parts: the small part associated with this partial solution is evaluated and this branch will be cut off from the solution space; and the other part is not fully explored and constitute the later promising region and surrounding region. So, each time a partial solution is evaluated, a cut which cuts off the partial solution is added to the overall solution space to improve the efficiency of the algorithm.

9.4.2 Multi-attribute Decision-Making

For the second evaluation method, MADM (multi-attribute decision-making) is used. By using MADM, many trade-off attributes with different measuring criterion can be normalized in the single view of measure, so that each attribute will be compared. MADM not only can show the evaluation values of each alternative, but also lets us give subjective weight to the attributes, consequently helping seek an optimal solution. The MADM method is mainly used for solving problems which have many alternatives and attributes, thus aiming at fixing the order of preference of attributes for many alternatives. Due to this, an easy and a convenient method, an entropy method, is used to confirm, check, and evaluate the items. On the other hand, MADM can perform evaluations on the basis of not only the internal weights according to entropies, but also the weights that are directly inputted by the user. Weights of cost and time can be adjusted by the user, which will satisfy our second research gap in the literature survey.

In this method, two artificial alternatives are hypothesized: ideal alternative, negative ideal alternative Hwang (1981) and Rao (2007).

Step 1 Construct normalized decision matrix.
Step 2 Calculate the entropy value using the normalized decision matrix.
Step 3 The degree of divergence for each attribute is calculated.
Step 4 The more divergent the performance ratings, the higher are the corresponding degree of divergence, thus more important the attribute.
Step 5 Finally the objective weight for each attribute is calculated.
Select the Alternative with higher weights.

9.5 An Illustrative Example

The sea and air mode transport routes where the service providers are available for the order, which falls under the earliest final delivery date: 09/05/2011 and latest final delivery date: 15/05/2011 are shown in Fig. 9.1.

In the above network, every arc represents the link between two locations, every link has certain available service provider and every service provider has different capacity ranges and based on the range of capacity the transportation cost varies, this is taken care by the first objective function (minimization of cost), later transportation time varies only on service providers that is taken care by the second objective function (minimization of time). The order from customer and it is grouped based on same origin and destination are given in Tables 9.1 and 9.2.

Let consider the following three types of commodities in our problem.

Commodity 1 Garment and Textile.
Commodity 2 Consolidation (mixed with or without garment and textile).

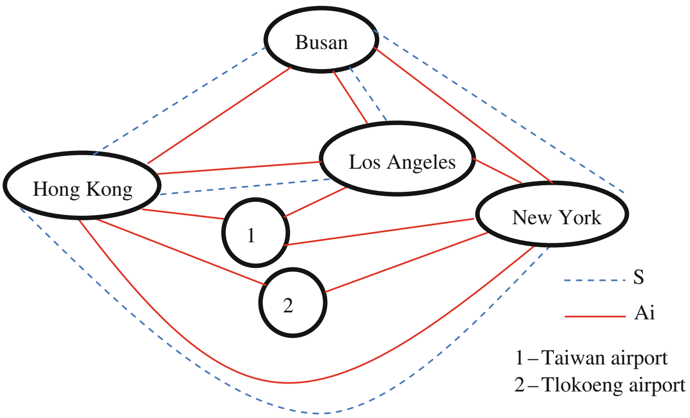


Fig. 9.1 Intermodal transportation network

Table 9.1 Customer orders

Order	C1 of 1,000 kgs has to be shipped from Hong Kong to Los Angeles
Order	C3 of 16,000 kgs from Busan to Los Angeles
Order	C2 of 5,000 kgs from Hong Kong to Los Angeles
Order	C2 of 3,000 kgs from Busan to New York
Order	C2 of 5,000 from Busan to New York

Table 9.2 Orders grouped based on O–D and commodity

Orders	Origin	Destination	Capacity
G1	Hong Kong	Los Angeles	6,000 kgs
G2	Busan	Los Angeles	16,000 kgs
G3	Busan	New York	8,000 kgs

Commodity 3 Straight Loads only of other item not listed Groups 1 and 2, but excluding cargo requiring special equipment and dangerous/hazardous cargo.

Taking every group separately and selecting the best route for it in the intermodal network based on our solution techniques, to start with it, take group one and get the set of feasible routes using nested partition method.

Step 0 Consider the entire solution as sub feasible region.

Step 1 With the source node as Hong Kong, obtained the next feasible region from the sub region, shown in Fig. 9.2.

Step 2 Calculating the promising index, for each region σ_j, $j = 1, 2, \ldots, M + 1$, calculate the promising index as the best performance value within the region: $I(\sigma_j) = \min_{i=1,2,\ldots,N_j} f(x_i^j)$, $j = 1, 2, \ldots, M + 1$. Since it had only two regions, it is easily calculated.

Step 3 Makes another partition from the feasible region now again check the promising index, this time, cross check it with not only the current sub feasible region but also the past sub feasible regions, if found sub region more promising than feasible, it has to be brought under feasible region and further evaluated.

In the next iteration, see the predecessor in the feasible region toward the destination, if it does not fall under feasible route and predecessor of sub feasible region falls into feasible route, then back track to the previous region and move the best into feasible set.

Step 4 No backtrack in this process because, the sub feasible region does not have a better promising index.

Step 5 Set of feasible routes is saved.

Same steps are continued for minimization of time. The best service providers cost and time in every link is shown in Fig. 9.3. The set of feasible routes from the nested partition algorithm are given in Table 9.3.

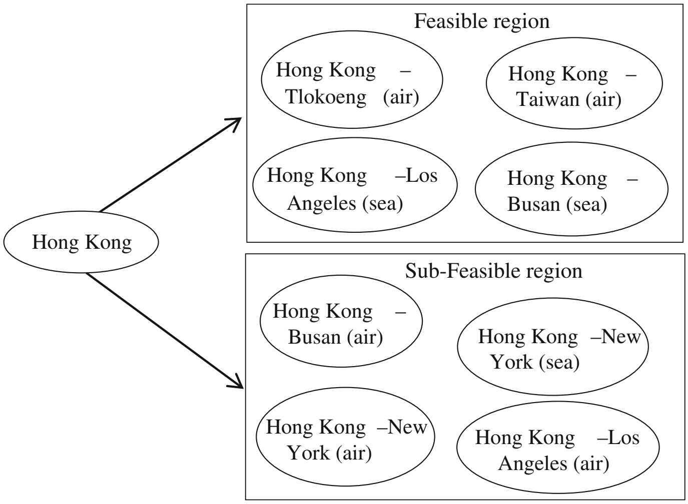


Fig. 9.2 The partition in nested partition method

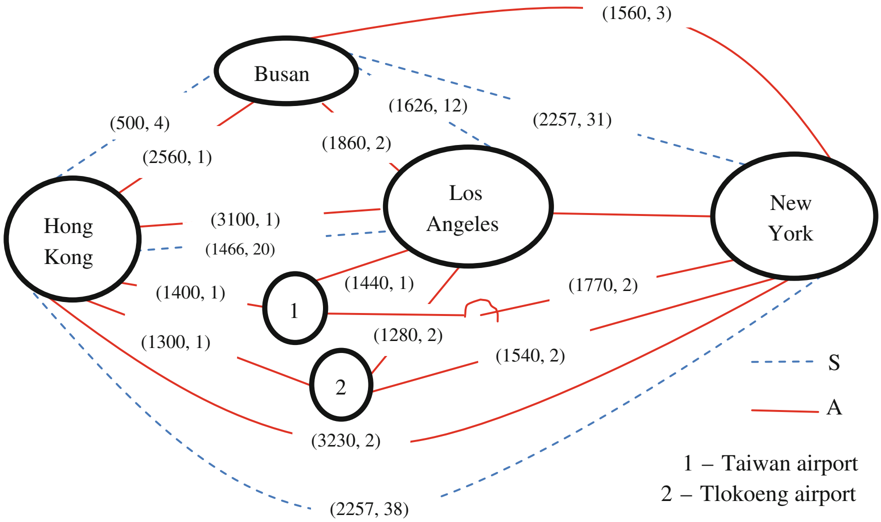


Fig. 9.3 Intermodal network with cost and time

With the above available feasible routes to select the best route as the customer preferred, use MADM evaluation, giving the weight (0–1) for cost and time based on the preference, the one with more preference will have a value close to 1 and the one with minimum preference will have preference close to zero. There are several multi-attribute decision-making methods, such as simple additive weighting (SAW) method, weighted product method (WPM), analytic hierarchy process (AHP) , and Technique for Order Preference by Similarity to the Ideal Solution (TOPSIS). Among these, TOPSIS is simple and gives almost near ideal results.

The result for giving equal weight to cost and time is given in Table 9.4, repeating the experiment by changing the weights thus, obtained different results for different weight conditions which are given in Table 9.5. Similarly, the analysis for other two groups and their results are given in Tables 9.6 and 9.7.

Thus from the result of nested partition method, the set of feasible routes are obtained, using our input to MADM evaluation, which gives the results, such as if a customer needs both minimum cost and minimum time then he must take the route of Hong Kong–(air)–Tlokoeng–(air)–Los Angeles, for group 1 orders. The optimal cost and time is $2580, 3 days. Other weights give other routes such as for cost weight 0.2 and time weight 0.8 Hong Kong–(sea)–Busan–(air)–Los Angeles. The optimal cost and time is $2360, 6 days. Thus, combining nested partition method and MADM gives us the best route in a very easy and simpler way. It also gives the flexibility of customer's choice to select their preferred route.

Group 2 has a capacity constraint in air mode routes; therefore, the result from nested partition is as follows (Tables 9.6 and 9.7).

Results for group 3 are carried out as same as group 1 and its final results are as follows, assuming that the customer's preferences are equal weight to cost and time (Table 9.8).

Table 9.3 Set of feasible routes from origin to destination

Origin–Destination	Cost, time
Hong Kong–(air)–Los Angeles	3100, 1
Hong Kong–(air)–Taiwan–(air)–Los Angeles	2840, 2
Hong Kong–(air)–Tlokoeng–(air)–Los Angeles	2580, 3
Hong Kong–(air)–Busan–(air)–Los Angeles	4420, 3
Hong Kong–(sea)–Busan–(air)–Los Angeles	2360, 6
Hong Kong–(air)–Busan–(sea)–Los Angeles	4180, 13
Hong Kong–(sea)–Busan–(sea)–Los Angeles	2126, 16
Hong Kong–(sea)–Los Angeles	1466, 20

Table 9.4 Results for equal weightage to cost and time using MADM

Cost Wc = 0.5	Time Wt = 0.5	MADM evaluation
1,466	20	0.350183
2,126	16	0.369723
2,360	6	0.727644
2,580	3	0.806780
2,840	2	0.794504
3,100	1	0.775311
4,180	13	0.316152
4,420	3	0.619702

Table 9.5 Result with different weight

NP results–feasible routes (Cost, time)	MADM evaluation		
	Wc = 0.2, Wt = 0.8	Wc = 0.7, Wt = 0.3	Wc = 0.9, Wt = 0.1
1466, 20	0.118728	0.557017	0.829062
2126, 16	0.229325	0.54381	0.737784
2360, 6	0.736105	0.712273	0.698912
2580, 3	0.884906	0.709983	0.632323
2840, 2	0.920698	0.664258	0.550335
3100, 1	0.930763	0.622519	0.47081
4180, 13	0.364149	0.225034	0.107034
4420, 3	0.839571	0.414891	0.155716

Table 9.6 NP results for group 2

Origin–Destination	Cost, time
Busan–(sea)–Los Angeles	1626, 12
Busan–(Sea)–Hong Kong–(sea)–Los Angeles	1966, 24

Table 9.7 MADM evaluation for group 2

Cost Wc = 0.5	Time Wt = 0.5	MADM evaluation
1,626	12	1
1,966	24	0

Table 9.8 Results for group 3

Origin–Destination	Cost Wc = 0.5	Time Wt = 0.5	MADM evaluation
Busan–(air)–New York	1,560	3	0.85361
Busan–(sea)–New York	2,257	31	0.42685
Busan–(sea)–Hong Kong–(sea)–New York	2,757	42	0.280343
Busan–(sea)–Hong Kong–(air)–Tlokoeng–(air)–New York	3,340	7	0.790033
Busan–(sea)–Hong Kong–(air)–Taiwan–(air)–New York	3,670	7	0.76225
Busan–(sea)–Hong Kong–(air)–New York	3,730	6	0.766761
Busan–(air)–Hong Kong–(sea)–New York	4,817	39	0.127221
Busan–(air)–Hong Kong–(air)–Tlokoeng–(air)–New York	5,400	4	0.661266
Busan–(air)–Hong Kong–(air)–Taiwan–(air)–New York	5,730	4	0.693263
Busan–(air)–Hong Kong–(air)–New York	5,790	3	0.645233

9.6 Conclusions

This study has suggested an algorithm to find the best route in the international intermodal freight transportation network. This international intermodal freight routing is complicated by following essential characteristics, (1) multi-objective : minimization of travel time and travel cost, (2) schedules and delivery times of every service provider in each pair of location, (3) variable cost must be included in every location. A MILP model has been presented first, since it is a NP-hard problem, it follows the proposed algorithm that is heuristic. Based on nested partitions method, one of the recent successful algorithms which is widely used for large-scale optimization problems. Results of the nested partitions method are inputted to MADM model, which evaluates the feasible solutions and gives out the optimal solution. The feasibility our proposed algorithm is tested by applying it to an illustrative experiment, which then effectively and efficiently solves and provides the best optimal route.

In the future work, we will include the terminal connections and link this algorithm into a user interface. This algorithm can be used to develop a network planning system, network optimization system, or decision support system and it can be equipped with any third party logistics company.

References

Barnhart C, Ratliff HD (1993) Modelling intermodal routing. J Bus Logis 4(1):205–223

Boardman BS, Malstrom EM, Butler DP, Cole MH (1997) Computer assisted routing of intermodal shipments. Comput Ind Eng 33(1–2):311–314

Bookbinder JH, Fox NS (1998) Intermodal routing of Canada–Mexico shipments under NAFTA. Transp. Res. Part E 34(4):289–303

Çakır O (2009) Benders decomposition applied to multi–commodity, multi–mode distribution planning. Expert Syst Appl 36(4):8212–8217

Chang TS (2008) Best routes selection in international intermodal networks. Comput Oper Res 35(9):2877–2891

Cho JH, Kim HS, Choi HR (2012) An intermodal transport network planning algorithm using dynamic programming—a case study: from Bussan to Rotterdam in intermodal freight routing. Appl. Intell 36(3):529–541

Crainic TG, Rousseau JM (1986) Multi–commodity, multimode freight transportation: a general modeling and algorithmic framework for the service network design problem. Transp Res Part B 20(3):225–242

Hu ZH (2011) A container multimodal transportation scheduling approach based on immune affinity model for emergency relief. Expert Syst Appl 38(3):2632–2639

Hwang CL, Yoon KS (1981) Multiple attribute decision making: methods and applications. In: Lecture notes in economics and mathematical systems (Multiple Attribute Decision Making), vol 186. Springer, Berlin

Kang K, Niu H, Zhu Y, Zhang W (2010) Research on improved iIntegrated optimization model for mode and route in multimodal transportation basing on the PSO-ACO. Conf Publ Logistics Syst Intell Manage 3:1445–1449

Leung LC, Hui YV, Wang Y, Chen C (2009) A 0–1 LP model for the integration and consolidation of air cargo shipments. Oper Res 57(2):402–412
Min H (1991) International intermodal choices via chance–constrained goal programming. Transp Res Part A 25(6):351–62
Olafsson S (2003) Two–stage nested partitions method for stochastic optimization. Meth Comput Appl Prob 6(1):5–27
Rafay I, Charles RS (2010) Intermodal logistics: the interplay of financial, operational and service issues. Transp Res Part E 46(6):926–949
Rahim A, Farhad S (2011) Time–dependent personal tour planning and scheduling in metropolises. Expert Syst Appl 38(10):12439–12452
Rao V (2007) Decision making in the manufacturing environment using graph theory and fuzzy multiple attribute decision making methods. Springer Series in Advanced Manufacturing, doi: 10.1007/978-1-4471-4375-8_1. Springer-Verlag, London 2013
Shi L, Olafsson S (2000) Nested partitions method for global optimization. Oper Res 48(3):390–407
Shi L, Olafsson S (2008) Nested partitions method, theory and applications. Springer, Berlin
Shi L, Olafsson S, Sun N (1999) New parallel randomized algorithms for the travelling salesman problem. Comput Oper Res 26(4):371–394
Southworth F, Peterson BE (2000) Intermodal and international freight network modelling. Transp Res Part C 8(1–6):147–66
Tadashi Y, Bona FR, Jun C, Eiichi T (2009) Designing multimodal freight transport networks: a heuristic approach and applications. Transp Sci 43(2):129–143
Yang Y, Low MWJ, Tang LC (2011) Analysis of intermodal freight from China to Indian Ocean: a goal programming approach. J Transp Geog 19(4):515–527
Zhang Y, Jun Y, Wei G, Wu L (2010) Find multi–objective paths in stochastic networks via chaotic immune PSO. Expert Syst Appl 37(3):1911–1919

Chapter 10
An Evolutionary Algorithm with Path Relinking for a Bi-objective Multiple Traveling Salesman Problem with Profits

N. Labadie, J. Melechovsky and C. Prins

Abstract This chapter deals with a bi-objective multiple traveling salesman problem with profits (BOMTSPP), generalizing the classical TSP with profits (TSPP). The TSPP is in fact a generic name for TSP problems taking into account the length of the tour and profits collected at customers. However, all these problems are not really bi-objective: the two criteria are aggregated into a single objective or one of them is replaced by a constraint. Our BOMTSPP aims at building *m* cycles covering a subset of potential customers so that the total collected profit is maximized and the overall traveling distance is minimized. This new problem generalizes the TSPP in two directions: a true bi-objective treatment and the construction of multiple tours. The proposed solution method is an effective evolutionary algorithm, reinforced by a post-optimization procedure based on path-relinking (PR).

Keywords Multiple traveling salesman problem · Traveling salesman problem with profits · Multi-objective optimization · Evolutionary algorithm · Path-relinking

10.1 Introduction

Multi-objective combinatorial optimization (MOCO) is an important research field with a fast development over the last two decades. In many practical applications of combinatorial optimization problems, decision makers must deal with multiple

N. Labadie (✉) · C. Prins
ICD/LOSI, Université de Technologie de Troyes, 12 rue Marie Curie
CS 42060 10 004 Troyes Cedex, France
e-mail: nacima.labadie@utt.fr

J. Melechovsky
Department of Econometrics, University of Economics, Prague
náměstí W. Churchilla 4 130 67 3 Prague, Czech Republic

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_10,

and often conflicting criteria. The purpose of MOCO is thus to provide a set of nondominated solutions in a Pareto sense, rather than optimizing one objective or a weighted sum of objectives and providing a single solution.

This chapter presents the first study on a bi-objective multiple traveling salesman problem with profits (BOMTSPP), a generalization of the traveling salesman with profits (TSPP). The TSPP is a generic name for a family of single-vehicle routing problems where each customer must be visited at most once. Two performance criteria are taken into account: the length of the tour, like in the classical TSP, and profits which can be collected at customers. These two criteria are obviously conflicting. Minimizing tour length leads to less customer visits and reduces the total profit while maximizing the total profit instigates the traveler to visit more customers and increase tour length. Traveling salesman problems with profits are frequent in service activities. Among practical applications can be mentioned scheduling a traveling salesman to visit the most profitable customers (Tsiligirides 1984) or an intelligent tourist guiding system proposing to the user a subset of the most interesting touristic sites that can be visited within a given stay, see Schilde et al. (2009); Vansteenwegen et al. (2009a); Wang et al. (2008).

Despite the bi-objective nature of traveling salesman problems with profits, research mostly focused on single-objective variants, found in the literature under various names. Feillet et al. (2005) survey all these variants and propose a classification into three generic problems, depending on how the two criteria are treated.

- Both criteria are expressed in the objective function, by minimizing the travel costs minus the collected profit. This version is referred to as the profitable tour problem (PTP).
- The total profit is maximized while the length of the tour is limited to a given value $l_{\max}$. This version is called the orienteering problem (OP), the selective traveling salesman problem (STSP) or the maximum collection problem.
- Conversely, the travel costs are minimized but the collected profit must not be less than a given constant $p_{\min}$. This version is called the prize-collecting traveling salesman (PCTSP) or the quota traveling salesman problem (QTSP).

All these traveling salesman problems with profits are NP-hard. For an exhaustive survey on formulations and resolution methods developed for this family of problems, see, for instance, the recent paper of Vansteenwegen et al. (2011).

The OP is probably the most investigated variant. It was introduced by Tsiligirides in 1984 and consists in building a tour originating at a depot-node and visiting a subset of customers, without exceeding a given length. This length corresponds, for instance, to the range or maximum working time of the vehicle. A given profit is associated with each customer and is realized when the customer is visited. The objective is to maximize the total profit collected. An extension of the OP is its version with multiple tours, where a fixed number $m > 1$ of tours has to be built, each tour respecting the length limit $l_{\max}$. This problem was formulated by Chao et al. (1996) as the team orienteering problem (TOP).

Almost all resolution methods developed to tackle the TOP are heuristics. Among them we can cite a tabu search developed by Tang and Miller-Hooks (2005), a variable neighborhood search and a tabu search proposed in Archetti et al. (2007), an ant colony optimization method described by Ke et al. (2008) and a memetic algorithm presented in Bouly et al. (2010).

Recently, the TOP with time windows (TOPTW) has received much attention. Several metaheuristics are available for this problem where the service at a customer must start in a given time window: Montemanni and Gambardella (2009) proposed an ant colony system while Vansteenwegen et al. (2009b) developed an iterated local search. A hybrid evolutionary local search was studied by Labadie et al. (2010, 2011). The same authors (Labadie et al. 2012) designed a variable neighborhood search based on linear programing. An exact approach (branch-and-price) was presented by Boussier et al. (2007) for the TOP and a selective vehicle routing problem with time windows (SVRPTW). The latter is in fact a multivehicle extension of the STSP with time windows, with additional capacity constraints on the routes. Tricoire et al. (2010) introduced the multiperiod orienteering problem with multiple time windows (MuPOPTW), which generalizes the TOPTW over a multiperiod horizon. Each customer is associated with a fixed number of services and one specific time window for each period.

As already mentioned, the references cited up to now are not true multi-objective approaches. A multi-objective version of the TSPP was considered for the first time by Keller and Goodchild (1988). We are aware of only four other papers.

Riera-Ledesma and Salazar-González (2005) studied the traveling purchaser problem, in which the nodes represent markets for different products. The traveling purchaser must visit a subset of markets in order to purchase the required quantity of each product while minimizing the travel cost and the purchase cost.

In Jozefowiez et al. (2008), a multi-objective evolutionary algorithm including a local search based on ejection chains was developed to generate efficient solutions to the traveling salesman problem with profits. Bérubé et al. (2009a) designed an exact ϵ-constraint method for the same problem and, finally, Schilde et al. (2009) studied a new bi-objective variant of the OP, in which each customer has two different kinds of profits. The two objective functions considered are the maximization of both collected profits. The authors proposed an ant colony algorithm and a variable neighborhood search, both hybridized with a path relinking (PR) method, in order to generate Pareto-optimal solutions.

The BOMTSPP addressed in this chapter consists in building m cycles covering a subset of customers, to maximize the total collected profit and minimize the overall traveling distance. This new problem generalizes the TSPP in two directions: multiple tours and a true bi-objective treatment, with the determination of a set of nondominated solutions. We propose a multi-objective evolutionary algorithm to solve it. A path-relinking scheme is applied as a post-optimization process to obtain more nondominated solutions. The chapter is organized as follows. Section 10.2 defines the problem. In Sect. 10.3, the components of the multi-objective evolutionary algorithm are detailed. The path relinking is explained in Sect. 10.4. Numerical results are presented in Sect. 10.5 and 10.6 concludes the chapter.

10.2 Problem Description

Our BOMTSPP is defined on a complete undirected graph $G = (V, E)$ with a node-set $V = \{0, 1, 2, \ldots, n\}$ and an edge-set E. Node 0 is a special node called depot while the other vertices correspond to potential customers. A nonnegative profit p_i is associated with each customer $i = 1, 2, \ldots, n$. Each edge $e = [i, j]$ is associated with a travel time c_e. It is assumed that these times satisfy the triangle inequality. A set of m vehicles are available at node 0 to visit the customers. The profit of each customer can be collected at most once.

The problem consists in building m tours starting and ending at the depot 0, such that the total profit is maximized and the total travel time is minimized. For a given solution S, these two objective functions are denoted as $f_1(S) = \sum_{i \in V} p_i y_i$ and $f_2(S) = \sum_{e \in E} c_e x_e$. The binary variable y_i is equal to 1 if and only if customer i is visited in solution S. The integer variable $x_e \in \{0, 1, 2\}$ indicates the number of times edge e is traversed. The value 2 corresponds to a direct trip to a customer j: in that case edge $[0, j]$ is traversed twice.

10.3 Multi-objective Evolutionary Algorithm

Evolutionary algorithms (EA) have received considerable attention and are well adapted to solve multi-objective optimization problems. This is due to the fact that these are population-based approaches and also because it is relatively easy to modify a single-objective evolutionary algorithm to find a set of nondominated solutions in a single run. There are two issues that require a particular attention: the evaluation of the fitness function and the diversity of the population. A survey on multi-objective EA can be found for instance in Konak et al. (2006).

The algorithm developed to address the problem under study is based on the nondominated sorting genetic algorithm version 2 (NSGA-II) proposed by Deb et al. (2002). This method uses a ranking based on a nondominated sorting algorithm for the fitness assignment and a crowding distance for maintaining the diversity of the population. The crowding distance gives an estimation of the population density around a solution. The advantage of using the crowding distance is that it does not require any user-defined parameter. Therefore, there is only one single parameter to be defined for NSGA-II: the population size, denoted as

N in the sequel. Our implementation is reinforced by a local search (hence, it can be considered as a bi-objective memetic algorithm) and a path relinking process.

10.3.1 Solution Encoding and Initial Population

Each solution in the population is encoded as a chromosome defined by an ordered sequence containing the customers visited, without depot copies to delimit the trips. As some customers can be unvisited, chromosomes can have different lengths. The ones obtained by crossovers are split to give a BOMTSPP solution, using a procedure explained in Sect. 10.3.5. Conversely, a solution is easily converted into a chromosome by concatenating the lists of customers of its trips: for instance, a solution with two trips (0, 3, 7, 0) and (0, 4, 1, 5, 0) corresponds to chromosome (3, 7, 4, 1, 5) when the occurrences of the depot-node are removed.

Two extreme solutions are first included in the initial population. The first solution is composed of the m customers closest to the depot, each customer being visited by a direct trip. Clearly, this first solution is minimal for the second objective f_2 (total travel time). Then, all the other customers are inserted one by one in this solution, such that the increase in length of each insertion is minimal. This gives the second solution of the population. As all customers are visited, all the profits are collected and the first objective f_1 (total profit) is clearly maximized.

The remaining $N-2$ initial solutions give priority either to the first or second objective, converting the other into a constraint. The solutions in which we try to maximize the total profit, like in the TOP, are called TOP solutions. The ones in which we try to minimize travel time correspond to a multivehicle version of the prize-collecting traveling salesman problem, they are called m-PCTSP solutions for this reason.

10.3.1.1 Construction of TOP Initial Solutions

In the initial population, $N_1 = (N-2)/2$ TOP solutions are generated by varying between two values l_{beg} and l_{end} the upper limit $l_{\max}$ on the duration of each route. The value of l_{beg} (resp. l_{end}) is the maximum duration of the routes in the first (resp. second) solution of the initial population. Starting from l_{beg} the duration limit for each TOP solution is increased by a step $l_{\mathrm{step}} = (l_{\mathrm{end}} - l_{\mathrm{beg}})/N_1$.

For a given $l_{\max}$, the TOP solution is constructed as follows. First, m direct trips are built with the m most profitable customers such that $l_{\max}$ is respected, that is, the ones reachable in a maximum time $l_{\max}/2$. Then, for each node i (customer or depot) in the partial solution and each unvisited customer j, a ratio $r_{ij} = p_i^2/\Delta_{ij}$ is computed, where $\Delta_{ij} = c_{ij} + c_{jk} - c_{ik}$ is the travel time variation and k is the successor of i in his trip. The feasible insertion which maximizes this ratio is executed. This process is repeated until no further insertion is possible.

10.3.1.2 Construction of the m-PCTSP Initial Solutions

The population is finally completed with $N_2 = (N-2)/2$ m-PCTSP solutions, by varying the lower limit $p_{\min}$ for the total profit collected between two values p_{beg} and p_{end}. The initial value p_{beg} corresponds to the sum of the m smallest customer profits while the final value p_{end} is the sum of all customers profits. The step size used is $p_{\text{step}} = (p_{\text{end}} - p_{\text{beg}})/N_2$.

The following heuristic is used to solve approximately the m-PCTSP for a given value of $p_{\min}$. We determine a compact subset of customers S_{pmin} with a total profit not smaller than $p_{\min}$ and build m tours covering these customers. The set S_{pmin} is obtained by solving a fractional knapsack problem (FKP) derived from the original problem. The n items considered in the FKP are the customers, their weights are the profits p_i while their values g_i represent a cost estimation if item i is selected. The estimation chosen is the average travel time from node i to the other nodes in $V : g_i = 1/(n-1) \cdot \sum_{j \in V \setminus \{i\}} c_{ij}$. Then by setting the capacity of the knapsack to $p_{\min}$, an FKP instance is entirely defined and its resolution is equivalent to finding a subset of customers S_{pmin} such that the sum of their cost estimations is minimized and their total profit is at least $p_{\min}$.

The optimal solution of the FKP can be easily determined by a greedy algorithm (Dantzig 1957). Consider the items sorted in ascending order of utilities g_i/p_i:

$$\frac{g_{i_1}}{p_{i_1}} \leq \frac{g_{i_2}}{p_{i_2}} \leq \cdots \leq \frac{g_{i_n}}{p_{i_n}}$$

and:

$$t = \min\left\{ j = 1, 2, \ldots, n \middle| \sum_{k=1}^{j} p_{i_k} \geq p_{\min} \right\}$$

As all utilities are strictly positive, the optimal solution to FKP consists of items 1 to t in the sorted list of items.

Once the FKP solution has given the set S_{pmin}, m direct trips are built using the customers closest to the depot in S_{pmin}. The remaining customers of S_{pmin} are progressively added to the trips using the best insertion criterion, and the resulting solution is improved by local search.

All solutions in the initial population are improved via the local search procedure described in the next section. They are converted into chromosomes by concatenating the lists of customers of each trip. To favor a better diversity, if several solutions with the same cost are present in the population, only one is kept. If less than N solutions are obtained, the population is completed by randomly generated chromosomes. A random chromosome is obtained by setting first its length q which is randomly chosen in the interval $[m, n]$; where m is the number of vehicles and n is the number of potential customers. The q customers to include in the chromosome are then selected randomly.

10.3.2 Improvement Procedures

A local search $LS(k)$ based on four moves is used to improve the initial solutions and the new solutions generated by the crossover operator.

The three first moves that we call routing moves are the well-known 2-Opt, Or-Opt, and Exchange moves. They aim at reducing the total travel time without increasing the maximum duration of the trips ($l_{\max}$) nor changing the subset of visited customers. The 2-opt move replaces two edges in a route by two other edges. The Or-opt move removes a string of k consecutive customers to reinsert it at a different location. The Exchange move exchanges two strings of k customers. The Or-opt and Exchange moves may involve one or two trips.

The fourth and last move is called external move: a sequence of k customers is removed from a tour and replaced by a sequence of entering customers. This move is intended to change the subset of visited customers and, as its evaluation is more time-consuming, it is attempted by the local search only when no routing move is productive. The entering sequence is determined heuristically: unvisited customers are added one by one and the customer i added at each step is the one maximizing the ratio r_{ij}, already explained in Sect. 10.3.1.1.

We designed two variants of the local search, which differ in the implementation of the fourth move. In the variant called $LS1$, we try to increase the total profit f_1 like in the TOP and search for external moves maximizing the difference between the total profit inserted and the total profit removed, while respecting the maximum trip duration $l_{\max}$. In the variant $LS2$, the goal is to reduce the total travel time f_2, like in the m-PCTSP, without decreasing the total profit $p_{\min}$. The values of $l_{\max}$ and $p_{\min}$ are the ones observed in the input solution.

Finally, $LS1$ (resp. $LS2$) are embedded in a variable neighborhood descent (VND) called VND1 (resp. VND2). Starting with $k = 1$, this VND calls $LS(k)$. In case of improvement, k is reset to 1, otherwise it is incremented. The VND ends when reaching a maximum string length $k_{\max}$ without improvement. VND1 is applied to the TOP solutions of the initial population, VND2 to the m-PCTSP solutions, and both procedures are called to improve new offspring solution obtained by crossover.

10.3.3 Population Management

Nondominated sorting gives a rank q_S to each individual S in the population. The nondominated solutions form a subset F_1 such that $q_S = 1$. Then the next subset F_i is formed by solutions dominated only by subsets $F_1, F_2, \ldots, F_{i-1}$, and so on. The density of solutions surrounding an individual in the population is estimated by the crowding distance metric. This distance is calculated as the surface of the rectangle formed by the two nearest neighbors surrounding an individual. The population is first sorted according to the rank of individuals in the ascending order and then each subset is sorted in descending order of crowding distance. An archive A of nondominated solutions encountered during the search is kept aside.

The parents are selected by a binary tournament method. Two individuals are randomly selected. The one with lower rank is preferred and, if the two solutions have the same rank, the one with greater crowding distance is selected. This gives the first parent. This process is repeated to get the second parent.

Alogorithm 10.1 General framework of our NSGA-II variant

A:= ∅
t:= 0
Initialize P_t
Repeat
Find all non-dominated fronts F_i in P_t
Update the archive A of non-dominated solutions
Calculate the crowding distance
Sort each F_i of P_t in descending order of crowding distance
Reset the size of P_t to N
Repeat
Select two chromosomes with the binary tournament
Apply crossover, giving two children s1 and s2
Apply VND1 and VND2 to s1 and s2 with
a small probability π, giving s3 and s4
Add s3 and s4 to P_t
Until the population size is doubled
$t := t+1$
Until (t = MaxIter)

The general structure of the implemented evolutionary algorithm is illustrated in Algorithm 10.1. First the population P_t of N individuals is initialized as explained in Sect. 10.3.1. The nondominated sorting procedure then determines all nondominated fronts F_i in P_t. The crowding distance is calculated for each solution of the population P_t and each nondominated front F_i is sorted in descending order of the crowding distance. The size of P_t is reset to N which allows pruning away the solutions with the worst ranking. Finally, a new population is generated by performing the crossover and the local search.

10.3.4 Crossover Operator

The *Linear Order Crossover* (LOX) is used to combine two chromosomes. First, the length of the offspring (C_1 in Fig. 10.1) is set to n (the number of potential customers) and two crossover points are randomly selected in the first parent P_1. The customers positioned between the points in the first parent are copied into the offspring in the same positions and these customers are deleted from the second parent. The offspring is then completed by the remaining customers in parent P_2 taken in the same order from left to right.

As mentioned before, the chromosomes are not all of the same size. For this reason, when two parents are combined, the resulting offspring might contain

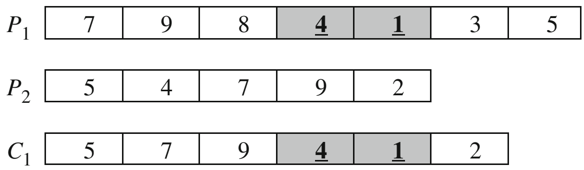


Fig. 10.1 Example of LOX crossover (nodes from P_1 underlined and in boldface)

empty spaces due to the different length of chromosomes representing the parents. These empty spaces must be removed by a left shift of the subsequent customers, and the length of the offspring must also be updated.

In the example of Fig. 10.1, the LOX crossover is performed on two chromosomes of different sizes. The first parent P_1 contains seven customers while the second parent P_2 only five customers. The cross section is defined by the sequence (4, 1) in P_1. The size of the produced offspring is 6.

The crossover operator generates two child chromosomes by altering the role of the parents. Each chromosome is converted into a real solution by using the split procedure explained in the next subsection. The resulting solution is further improved, with a probability π, by VND1 and VND2 and the improved solutions are added into the population if this does not create solutions with identical values of the objective functions. Once the population size gets doubled, the rank and the crowding distance are determined for each individual and the population is resorted. Finally, the size of the population is reset to its original size N by pruning away the N individuals with the worst ranking.

10.3.5 Split Procedure

The solution corresponding to a chromosome is evaluated by the split procedure originally designed by Prins (2004) for decoding a solution to the vehicle routing problem. The main advantage of the split procedure is its ability to determine an optimal solution with respect to the order of customers given by a sequence Q. The procedure determines the shortest path using m arcs in an acyclic auxiliary graph H consisting of a dummy vertex 0 and $|Q|$ customer vertices. An arc (i,j) in the graph H represents a tour containing a subsequence of customers $v_{i+1},\ldots,v_j$. The weight associated with the arc is the total traveling time of the tour. The shortest path in H can be determined with the Bellman's algorithm for directed acyclic graphs. At most for each customer $v_i, i=1,\ldots,n$, $m-1$ labels are assigned. Each label L_{ik} represents the shortest path from v_0 to v_i containing exactly k arcs. The optimal solution with respect to the sequence Q is then given by the label L_{nm} which indicates the shortest path from v_0 to v_n of exactly m arcs.

An example is illustrated in Figs. 10.2, 10.3, 10.4. The sequence of customers Q is depicted as the giant tour in Fig. 10.2. The shortest path from 0 to $v_n = e$ (bold lines in Fig. 10.3) has length 160 and the corresponding solution is depicted in Fig. 10.4.

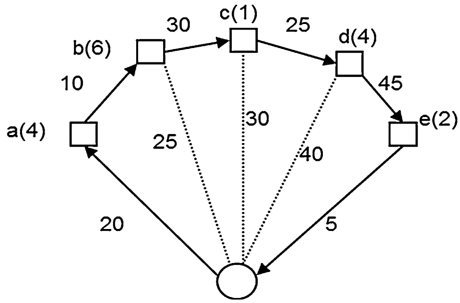


Fig. 10.2 The giant tour

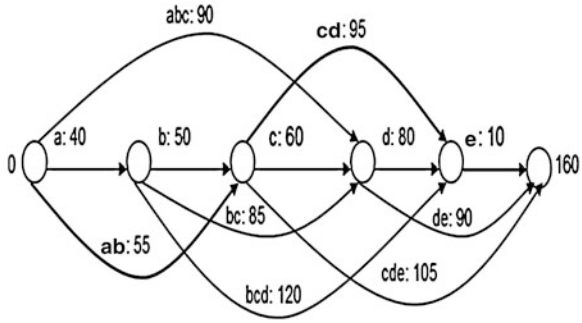


Fig. 10.3 The auxiliary graph *H*

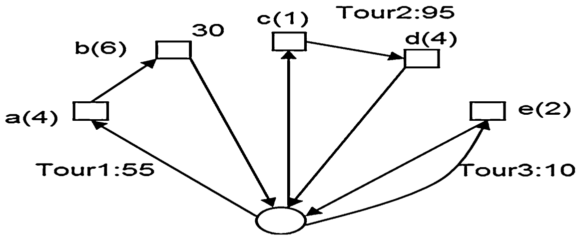


Fig. 10.4 The solution for $m = 3$

10.4 Path Relinking

Path relinking is an evolutionary search strategy which explores the trajectories connecting two solutions. The search starts in an initiating solution (starting solution) and it is further guided toward a guiding solution while several intermediate solutions are generated along the trajectory linking them. Each intermediate solution is generated by incorporating more attributes of the guiding solution. PR usually operates on a set of high quality (elite) solutions. See, for example, Glover et al. (2000) for more details.

Most of the research related to PR concerns single objective optimization problems. However, some applications of PR to multi-objective optimizations problems can be found in Basseur et al. (2005); Beausoleil et al. (2008); Jaszkiewicz and Zielniewicz (2009); and Martí et al. (2011).

10.4.1 Distance Measure

An important issue of a PR implementation is the definition of the distance measure between two solutions. Typically, the distance is defined as the minimum number of some basic operations required to convert one solution into another. Our distance measure extends that proposed by Martí et al. (2005) for permutation problems. It considers the relative position of an element rather than the absolute position. Given two permutations $P = (p_1, p_2, \ldots, p_n)$ and $Q = (q_1, q_2, \ldots, q_n)$, the permutation distance $d_M(P, Q)$ as defined by Martí et al. (2005) is computed as the number of times p_{i+1} does not immediately follow p_i in Q. In other words, it counts the number of times a pair of consecutive elements in P is broken in Q.

Such a distance measure suits well for permutations. For the case of two arbitrary sequences, two more operations are required: the deletion and the insertion of an element. To illustrate the modified distance measure, we defined $P(Q)$ to denote the ordered sequence obtained by considering the elements of P which are present in Q. Similarly, $Q(P)$ denotes the ordered sequence of elements in Q present in P. Furthermore, $l(P, P(Q)) = |P - P(Q)|$ denotes the number of elements in P which are not present in Q and $l(Q, Q(P)) = |Q - Q(P)|$ denotes the number of elements in Q not present in P. We define the distance between P and Q as follows:

$$D = d_M(P(Q), Q(P)) + l(P, P(Q)) + l(Q, Q(P))$$

The example in Fig. 10.5 illustrates the distance between P and Q. Both sequences share four vertices {4, 5, 7, 9}. After removing the excessive vertices from P {1, 3, 8} and Q {2}, we obtain three subsequences (7, 9), (4, 5), and (5, 4). As proposed in Labadie et al. (2008), the distance is reversal-independent. A breakpoint is counted for a pair of vertices (u, v) in P only if neither (u, v) nor (v, u) is found in Q. Hence, only one broken pair is counted and the total distance between P and Q equals 5.

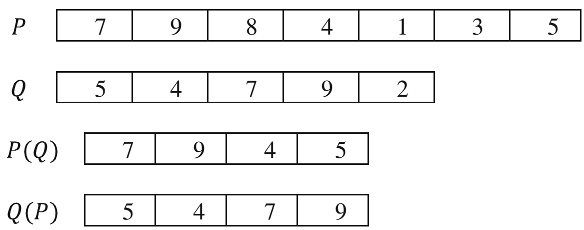


Fig. 10.5 Example illustrating the distance proposed ($D = 5$)

10.4.2 Generation of a Path

Given two solutions R and S with the distance $D(R, S)$, the basic operations are applied successively starting from R such that the distance to S is reduced. Performing an arbitrary operation at each step might lead to solutions of poor quality. To avoid this, the cost of each operation is evaluated and the least-cost one is performed. The resulting sequence is converted into a real solution with the *split* procedure.

If the solution is different from those in the archive, it is recorded into a list. Some solutions are improved with the local search procedure. The local search alternates between both strategies (VND1 and VND2) with a frequency of β iterations. This avoids finding the same local optima several times and reduces the computational time.

10.4.3 PR Implementation

PR is used as a post-optimization phase. It is applied to the archive of nondominated solutions generated by NSGA-II. The purpose is to explore the trajectories connecting two solutions in the objective space. The solutions in the archive are first sorted in ascending order according to f_1 (the profit). The initiating solution R is firstly taken from the top of the list. The guiding solution R is selected as its direct neighbor solution in the sorted list. For the next iterations, the guiding solution becomes the initiating one and the process is repeated until reaching the second extreme solution of the approximation front.

10.5 Computational Results

10.5.1 Test Instances

The computational experiments were conducted on two different data sets. The first set consists of TOP instances proposed by Chao et al. (1996). The authors have modified the instances generated originally by Tsiligirides (1984) for the single tour OP. There are seven data sets in total with the number of vertices equal to 21, 32, 33, 64, 66, 100, and 102, respectively. The starting and the ending points are assumed to be distinct in these instances. The problems within each data set differ in the maximal duration of the tour and in the number of tours. The first constraint is of course neglected since we deal with a bi-objective multiple TSP with profits. The number of tours ranges from 2 to 4.

The second dataset was proposed by Bérubé et al. (2009b) for the PCTSP. The instances are based on TSP instances from the TSPLIB due to Reinelt (1991).

Bérubé et al. (2009b) considered instances with n (the number of customers) varying from 70 up to 532. The customer profits were generated in three different ways. In the first case, the profit was set to 1 for all customers. For the second type, pseudo-random numbers are used and the profits of the third type are determined as a function of the travel distance between the customer and the depot. As in the first set of instances, the number m of tours ranged from 2 to 4.

10.5.2 Analysis of Results

Computational experiments were carried out on a PC equipped with 2.9 GHz dual core processor and 2 GB of RAM. The algorithms were coded in Delphi, a Pascal-like programming environment.

To the best of our knowledge, there are no reference results for the bi-objective multiple TSP with profits available in the literature. We have, therefore, adopted two common indicators to estimate the quality of the Pareto front approximation. The first one is the hyper-volume indicator or S metric proposed by Zitzler (1999). This indicator measures the volume in the objective space dominated by the potentially Pareto solutions. The second indicator is represented by the number of generated nondominated solutions. Furthermore, the contribution of the PR is evaluated by considering the number of solutions generated by PR, which dominate solutions generated by NSGA-II.

There are few parameters of the algorithm that need to be set up. The NSGA-II algorithm is basically controlled by the population size N, which was set to 100. The algorithm performed at most 3000 iterations and the complete algorithm (NSGA-II followed by PR) was terminated prematurely if a time limit of 1 h was reached. Local search procedure was applied to a child solution with the probability $\pi = 0.2$. The effectiveness of the local search procedure is strongly related to parameter $k_{\max}$ denoting the maximum number of customers in the sequence considered for a removal from the route. Larger values of $k_{\max}$ can deteriorate the actual solution with a minimal chance of finding a good entering sequence, since it is determined heuristically. It can also considerably increase the computing time. We have performed several tests for $k_{\max} = 2, 3, 4$ and 5 on a subset of 10 TSPLIB instances with profit type 3 and m ranging from 2 to 4. The best tradeoff between the quality of the approximate Pareto front and the computational time was achieved with $k_{\max} = 3$. When this parameter was set to five, the quality of the Pareto front approximation became worse of 0.5 % on average and the computational time increased on average by 7.1 %. With $k_{\max} = 4$ there was a small gain of 0.01 % in the quality of solutions while the computational times increased by 5 % on average. On the other hand, lower values of $k_{\max}$ lead to worse approximation of the Pareto front without any significant gain in computing times. As mentioned before, the quality of the Pareto front approximation was measured in terms of the hyper-volume indicator.

Only one parameter was applied to control the PR search process. It was the frequency β of calls to a local search procedure during the solution path exploration. This value was set to 3, that is, the local search was applied to every third generated intermediate solution.

Table 10.1 presents the results obtained for instances of Tsiligirides (1984) and Chao et al. (1996). The first two columns indicate the number of nodes and the number of tours, respectively. Columns 3–8 show results for each method. First it is the number of nondominated solutions in the Pareto front approximation (N_1 and N_2) obtained per data set. The two columns named *Hyp* presents the hyper-volume indicator. In the next column are reported computational times spent by each method. The last two columns of the table evaluate the contribution of PR to the final Pareto front approximation. N_{dom} represents the number of solutions of NSGA-II dominated by the Pareto front approximation obtained with PR. Contrarily, the last column shows the number of NSGA-II solutions which are not dominated by any PR solution.

The results indicate less important contribution of PR to the set of nondominated solutions on small instances with $n \leq 33$. However, with increasing n the importance of PR grows. For $n \geq 64$, the approximate Pareto front found by PR dominated on average 50 % of the original potentially Pareto solutions. The hypervolume was on average 1.5 % larger compared to the original Pareto front approximation.

Table 10.1 Results for instances of Tsiligirides (1984) and Chao et al. (1996)

n	m	NSGA-II			PR			N_{dom}	N_{eq}
		N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
21	2	53	0.490026	2.9	52	0.490232	0.1	2	51
32		44	0.420592	2.3	46	0.421426	0.0	2	42
33		68	0.550267	3.6	66	0.550990	0.1	3	65
64		101	0.458406	8.0	110	0.459521	2.5	14	86
66		138	0.492803	9.0	145	0.496431	5.2	46	92
100		139	0.533168	17.0	159	0.542198	17.7	97	42
102		163	0.607856	19.8	193	0.615750	21.5	127	36
21	3	49	0.462316	3.2	49	0.462316	0.1	0	49
32		34	0.386566	2.2	35	0.386710	0.0	1	33
33		61	0.530064	3.8	61	0.530631	0.1	2	59
64		98	0.367018	9.9	115	0.368971	6.2	30	68
66		84	0.468067	10.5	91	0.476909	6.3	33	51
100		121	0.487929	23.4	161	0.495096	17.8	82	39
102		133	0.590125	31.1	148	0.600902	28.5	96	37
21	4	48	0.435570	3.4	49	0.436503	0.1	2	46
32		38	0.362827	2.2	39	0.362897	0.1	0	38
33		60	0.514626	4.0	60	0.514626	0.0	0	60
64		90	0.286114	11.4	108	0.287458	5.4	28	62
66		68	0.455412	12.0	82	0.461378	2.2	32	36
100		110	0.431701	28.5	136	0.438571	20.9	61	49
102		98	0.581301	38.0	129	0.592749	33.8	71	27

Table 10.2 Average results obtained with NSGA-II and PR on TSPLIB instances

Profit type	m	NSGA-II			PR			N_{dom}	N_{eq}
		N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
1	2	95	0.509779	149.7	126	0.521764	519.3	66	30
	3	84	0.486192	344.6	110	0.500752	563.0	58	27
	4	80	0.471298	406.0	107	0.487141	405.2	56	25
2	2	127	0.535412	379.9	181	0.548695	527.7	89	38
	3	103	0.508222	628.0	147	0.523268	390.0	69	34
	4	96	0.489286	700.2	135	0.506264	352.4	66	30
3	2	121	0.487461	349.6	172	0.502200	461.6	86	35
	3	110	0.465920	485.5	151	0.481286	336.1	77	33
	4	108	0.454640	671.9	146	0.467698	273.5	69	39

Table 10.3 Results for TSPLIB instances with profit type 1 and $m = 2$, part I

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
st70		0.513433	8.5	53	0.524366	4.6	33	22
eil76	66	0.536981	9.6	68	0.541111	7.1	32	34
pr76	63	0.531173	11.8	60	0.540684	2.7	30	33
gr96	73	0.543292	15.3	70	0.550112	15.0	41	32
rat99	72	0.454731	16.3	71	0.460443	16.8	36	36
kroA100	70	0.485131	15.7	78	0.494118	9.5	39	31
kroB100	68	0.531057	15.6	77	0.537694	14.7	33	35
kroC100	62	0.474981	17.4	76	0.485223	15.3	38	24
kroD100	84	0.487086	17.3	77	0.498714	24.8	59	25
kroE100	72	0.513106	16.9	77	0.522280	14.6	30	42
rd100	83	0.520415	16.3	82	0.530196	17.5	51	32
eil101	85	0.550000	16.3	85	0.559327	26.7	66	19
lin105	62	0.506734	18.8	65	0.525916	11.4	35	27
pr107	91	0.466601	18.9	99	0.468631	7.1	31	60
pr124	56	0.502018	21.3	61	0.518894	15.4	37	19
bier127	104	0.698802	27.8	120	0.702238	65.2	67	37
ch130	86	0.537066	30.5	97	0.547404	51.4	62	24
pr136	94	0.475978	29.0	100	0.488797	70.4	70	24
gr137	92	0.488319	36.2	90	0.500853	13.1	53	39
pr144	92	0.463620	31.7	97	0.474333	10.3	35	57
ch150	111	0.500796	46.3	115	0.511634	147.3	94	17
kroA150	79	0.484529	47.7	106	0.499392	93.5	60	19
kroB150	80	0.501698	49.3	108	0.516823	61.3	63	17

NSGA-II spent on average 11.7 s per instance. PR required additional 8.0 s on average to improve the approximate Pareto front.

Table 10.2 summarizes the results obtained with NSGA-II and PR on the instances from TSPLIB. The summary is reported for each data set determined by

Table 10.4 Results for TSPLIB instances with profit type 1 and $m = 2$, part II

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
pr152	79	0.429651	30.6	97	0.445233	25.0	35	44
u159	68	0.493109	43.9	91	0.507923	61.1	48	20
rat195	97	0.476730	87.1	134	0.482481	110.7	49	48
d198	114	0.433898	79.0	128	0.439158	53.2	82	32
kroA200	120	0.508071	80.9	150	0.520103	255.9	97	23
kroB200	111	0.513100	92.0	140	0.522891	199.0	91	20
gr202	117	0.539799	82.7	145	0.549880	463.7	99	18
ts225	108	0.476742	105.5	133	0.488587	316.3	69	39
tsp225	98	0.458618	115.6	132	0.476359	298.4	81	17
pr226	76	0.484344	72.1	116	0.514596	39.5	43	33
gr229	109	0.625532	106.1	146	0.634561	377.2	88	21
gil262	102	0.486976	170.5	143	0.504710	692.1	84	18
pr264	88	0.460526	126.5	110	0.493835	412.6	76	12
a280	120	0.468316	153.5	178	0.481301	674.2	104	16
pr299	100	0.459295	201.8	154	0.477589	499.3	83	17
lin318	84	0.511602	263.8	146	0.528098	751.2	73	11
rd400	137	0.531833	539.5	251	0.544856	3173.3	126	11
fl417	92	0.519688	431.9	202	0.544039	943.4	75	17
gr431	173	0.682739	573.1	246	0.687304	3063.9	117	56
pr439	132	0.579506	482.3	216	0.590164	2267.7	109	23
pcb442	143	0.509601	567.0	287	0.520893	3048.8	117	26
d493	145	0.452928	748.3	261	0.463315	2968.4	93	52
att532	162	0.579689	1196.4	262	0.584083	2475.3	84	78

the type of the profit and the number m of tours as indicated in columns 1 and 2. The meaning of the remaining columns is the same as in Table 10.1.

The results show that PR was able to improve significantly the approximate Pareto front found by NSGA-II. The size of the Pareto front after PR has been applied increased by 38 % on average. This new approximation dominated on average almost 70 % of the original solutions and it contained only 22 % solutions from the original front. The average hyper-volume indicator was about 3 % larger for PR approximate Pareto fronts. NSGA-II spent on average 457.3 s while PR required additional 425.4 s on average to improve the approximate Pareto front. However, in the case of some difficult instances containing around 500 customers, PR was not executed completely due to the imposed maximum time limit.

Detailed results on TSPLIB instances are presented in Tables 10.3–10.20 . The tables follow the already described structure. The results prove that PR was able to improve significantly the approximate Pareto front for every instance. The PR front dominated at least 50 % of the original solutions. On the other hand, PR spent larger amount of computing time especially on complicated instances with more than 200 customers.

Table 10.5 Results for TSPLIB instances with profit type 1 and $m = 3$, part I

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
st70	47	0.497211	14.6	53	0.512243	5.2	5.2	22
eil76	65	0.510258	14.5	66	0.521174	14.3	14.3	25
pr76	54	0.500931	14.5	56	0.509794	9.8	9.8	31
gr96	63	0.526844	22.7	66	0.536999	34.7	34.7	32
rat99	58	0.423047	28.8	65	0.449138	26.9	26.9	23
kroA100	74	0.462181	22.7	74	0.472546	29.3	29.3	22
kroB100	59	0.500538	26.4	62	0.517602	25.1	25.1	24
kroC100	53	0.446529	27.1	59	0.469842	31.5	31.5	20
kroD100	56	0.464089	28.9	62	0.485809	25.0	25.0	21
kroE100	63	0.497554	24.5	71	0.511188	36.2	36.2	21
rd100	62	0.499967	28.8	69	0.516993	33.4	33.4	23
eil101	71	0.533083	25.5	81	0.544947	40.9	40.9	16
lin105	63	0.494004	28.1	69	0.507782	16.9	16.9	35
pr107	76	0.454617	28.7	80	0.460009	57.2	57.2	48
pr124	58	0.473177	36.8	65	0.499770	35.2	35.2	20
bier127	96	0.686603	50.1	101	0.692451	134.1	134.1	23
ch130	80	0.505789	45.8	84	0.519967	98.9	98.9	20
pr136	83	0.459020	49.5	94	0.473342	100.7	100.7	20
gr137	101	0.473777	56.4	93	0.480867	46.2	46.2	48
pr144	79	0.466332	57.1	86	0.476559	75.5	75.5	42
ch150	84	0.461554	92.8	88	0.480517	157.0	157.0	21
kroA150	84	0.468561	71.3	112	0.484041	141.3	141.3	14
kroB150	64	0.473354	93.2	88	0.491884	77.9	77.9	12

Table 10.6 Results for TSPLIB instances with profit type 1 and $m = 3$, part II

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
pr152	63	0.420013	68.6	88	0.437593	71.5	39	24
u159	79	0.448521	81.6	89	0.465200	123.2	63	16
rat195	95	0.468409	155.2	116	0.476224	520.2	47	48
d198	106	0.384585	136.6	126	0.388780	255.6	68	38
kroA200	89	0.480370	142.6	129	0.495200	344.8	75	14
kroB200	88	0.496640	134.5	117	0.511210	393.2	73	15
gr202	103	0.500306	157.8	122	0.511093	707.4	80	23
ts225	103	0.452510	218.9	119	0.472258	380.0	62	41
tsp225	76	0.429342	243.2	119	0.448752	650.6	71	5
pr226	93	0.461349	126.9	115	0.483039	300.9	77	16
gr229	111	0.593962	170.4	129	0.605913	1129.5	87	24
gil262	80	0.438377	365.5	121	0.462575	972.5	71	9
pr264	68	0.461005	207.0	126	0.492287	719.0	60	8
a280	100	0.437073	248.0	137	0.452208	1010.7	86	14

(continued)

Table 10.6 (continued)

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
pr299	108	0.456121	344.6	144	0.474892	1645.7	87	21
lin318	94	0.476729	484.8	153	0.501090	1791.0	85	9
rd400	142	0.513760	883.3	229	0.524269	2780.3	111	31
fl417	91	0.517060	985.6	186	0.533898	2518.8	77	14
gr431	143	0.654343	1643.8	206	0.658298	1975.6	80	63
pr439	123	0.566013	1084.2	218	0.578491	2570.8	101	22
pcb442	95	0.494304	1619.2	159	0.502124	1980.8	44	51
d493	127	0.394810	2098.8	179	0.399964	1544.1	54	73
att532	116	0.540199	3360.5	172	0.543749	259.4	38	78

Table 10.7 Results for TSPLIB instances with profit type 1 and $m = 4$, part I

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
st70	51	0.478569	14.3	50	0.497845	3.9	21	30
eil76	65	0.493036	18.0	57	0.504700	14.1	49	16
pr76	55	0.464418	16.9	52	0.477020	7.5	26	29
gr96	57	0.489571	31.2	61	0.504286	28.2	42	15
rat99	66	0.422294	30.6	72	0.431945	29.3	28	38
kroA100	62	0.448584	31.7	70	0.468849	33.3	51	11
kroB100	67	0.495951	31.8	60	0.517195	14.2	42	25
kroC100	49	0.435106	32.9	56	0.454791	7.8	31	18
kroD100	61	0.452469	38.5	73	0.464466	8.2	32	29
kroE100	56	0.482896	30.6	65	0.502250	27.8	42	14
rd100	57	0.482026	34.6	64	0.496671	21.9	27	30
eil101	67	0.515944	32.3	75	0.527760	55.7	47	20
lin105	63	0.472399	38.8	64	0.487734	23.0	37	26
pr107	71	0.449112	24.0	74	0.457263	32.5	22	49
pr124	62	0.460955	50.3	70	0.485215	13.5	50	12
bier127	83	0.683514	59.2	101	0.689036	153.2	57	26
ch130	79	0.490538	64.9	87	0.507029	98.6	64	15
pr136	64	0.433286	61.4	85	0.459026	71.8	52	12
gr137	96	0.451451	78.4	92	0.461555	39.9	61	35
pr144	75	0.455909	56.2	93	0.463276	45.0	32	43
ch150	70	0.468051	105.1	97	0.486942	116.4	51	19
kroA150	71	0.440381	89.0	78	0.470166	98.8	63	8
kroB150	65	0.439014	108.4	82	0.462273	65.4	53	12

Table 10.8 Results for TSPLIB instances with profit type 1 and $m = 4$, part II

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
pr152	66	0.403986	86.9	75	0.428039	31.9	49	17
u159	84	0.427865	95.6	89	0.447929	97.7	72	12
rat195	87	0.458964	194.4	113	0.469608	140.9	57	30
d198	111	0.347112	185.1	135	0.350597	219.5	69	42
kroA200	79	0.453479	244.6	101	0.482110	306.2	73	6
kroB200	96	0.474426	187.6	112	0.495754	286.6	83	13
gr202	89	0.467666	225.9	122	0.478417	573.3	68	21
ts225	83	0.436877	284.9	106	0.452324	236.8	57	26
tsp225	81	0.420941	285.8	121	0.443665	435.7	73	8
pr226	86	0.455003	189.6	130	0.471670	193.7	55	31
gr229	106	0.566200	264.0	120	0.581388	1183.8	90	16
gil262	86	0.426713	443.0	133	0.449345	827.4	76	10
pr264	79	0.482519	244.5	151	0.501586	210.7	64	15
a280	89	0.430838	336.3	147	0.444677	541.3	67	22
pr299	108	0.453066	492.2	148	0.474546	380.3	84	24
lin318	80	0.487227	688.7	156	0.505811	941.1	65	15
rd400	128	0.492451	1349.2	194	0.503412	2268.7	90	38
fl417	91	0.500603	910.0	184	0.527186	1380.2	80	11
gr431	129	0.638648	1743.1	198	0.644329	1929.0	80	49
pr439	108	0.567814	1438.1	210	0.584693	2298.0	94	14
pcb442	102	0.491239	2084.8	174	0.497199	1522.7	43	59
d493	105	0.362354	2140.1	169	0.366766	1483.0	51	54
att532	110	0.528255	3482.0	142	0.530164	140.0	33	77

Table 10.9 Results for TSPLIB instances with profit type 2 and $m = 2$, part I

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
st70	87	0.529662	11.8	93	0.541972	8.0	48	39
eil76	116	0.566036	14.1	130	0.574353	14.7	66	50
pr76	82	0.552798	15.4	90	0.562782	8.9	41	41
gr96	111	0.567931	25.3	122	0.584579	32.1	70	41
rat99	103	0.493287	30.5	105	0.506116	25.2	71	32
kroA100	98	0.525447	24.3	110	0.535944	27.7	55	43
kroB100	131	0.577324	25.6	153	0.584264	25.9	68	63
kroC100	104	0.524832	27.1	114	0.538730	31.5	67	37
kroD100	108	0.531864	27.2	135	0.540026	26.7	66	42
kroE100	119	0.562304	26.4	149	0.571848	34.3	74	45
rd100	112	0.572515	27.6	133	0.588894	34.6	79	33
eil101	113	0.587490	25.1	142	0.596576	41.3	74	39
lin105	90	0.531354	31.9	118	0.544565	13.1	52	38

(continued)

Table 10.9 (continued)

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
pr107	163	0.488453	38.2	204	0.495715	47.7	89	74
pr124	85	0.511399	41.7	116	0.526804	30.3	53	32
bier127	174	0.702432	48.2	277	0.708070	136.0	121	53
ch130	146	0.556665	45.8	162	0.570018	98.9	127	19
pr136	135	0.522389	57.1	167	0.537154	93.1	91	44
gr137	107	0.505343	75.1	138	0.521993	27.5	67	40
pr144	96	0.433070	64.5	104	0.458057	68.1	71	25
ch150	151	0.550184	68.2	218	0.560002	181.6	115	36
kroA150	136	0.531361	67.7	168	0.546103	144.9	112	24
kroB150	97	0.557263	72.8	155	0.568996	98.3	65	32

Table 10.10 Results for TSPLIB instances with profit type 2 and $m = 2$, part II

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
pr152	108	0.438800	68.2	153	0.454047	71.9	60	48
u159	90	0.504727	94.1	146	0.524511	110.7	66	24
rat195	149	0.513136	152.1	238	0.522815	523.3	124	25
d198	115	0.418673	163.2	160	0.439162	229.0	97	18
kroA200	141	0.541629	151.2	242	0.552074	336.2	102	39
kroB200	131	0.553670	149.3	214	0.566562	378.4	105	26
gr202	155	0.557050	170.6	194	0.570481	694.6	132	23
ts225	128	0.477532	175.1	164	0.492874	423.8	115	13
tsp225	113	0.484977	245.6	136	0.505101	648.2	93	20
pr226	101	0.506278	167.1	155	0.528037	260.7	62	39
gr229	188	0.644299	190.4	302	0.654309	1109.5	160	28
gil262	140	0.528733	305.7	227	0.545166	1032.3	113	27
pr264	119	0.489486	280.0	177	0.514367	646.0	102	17
a280	173	0.510852	324.4	319	0.523302	934.3	150	23
pr299	118	0.490595	647.7	199	0.512003	1342.6	101	17
lin318	149	0.537054	592.9	248	0.551685	1682.9	137	12
rd400	156	0.551602	1110.1	277	0.566078	2553.5	127	29
fl417	140	0.528257	900.3	297	0.551866	2604.1	109	31
gr431	249	0.697171	1163.8	405	0.701687	2455.6	137	112
pr439	146	0.584955	1335.6	272	0.600300	2319.4	114	32
pcb442	168	0.533965	1453.4	312	0.542034	2146.6	95	73
d493	167	0.470602	2003.6	211	0.479255	1639.3	93	74
att532	176	0.598780	2592.4	271	0.602029	1027.5	52	124

Table 10.11 Results for TSPLIB instances with profit type 1 and $m = 3$, part I

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
st70	85	0.513391	16.5	91	0.531015	6.0	44	41
eil76	90	0.542027	20.5	104	0.553006	12.4	54	36
pr76	80	0.517400	21.6	91	0.527586	10.7	41	39
gr96	90	0.539359	37.7	105	0.552581	32.3	63	27
rat99	104	0.477733	48.8	115	0.484288	19.7	52	52
kroA100	79	0.500287	34.8	93	0.517162	23.4	62	17
kroB100	78	0.550569	35.9	113	0.566995	21.3	47	31
kroC100	78	0.495932	44.8	98	0.508140	15.0	41	37
kroD100	91	0.512451	42.6	111	0.531440	25.5	68	23
kroE100	91	0.547199	41.9	109	0.561394	33.3	57	34
rd100	91	0.545281	44.3	115	0.563923	27.3	66	25
eil101	100	0.558804	38.2	131	0.570321	37.5	78	22
lin105	75	0.513090	47.0	107	0.527944	14.6	41	34
pr107	138	0.481938	56.5	174	0.486982	34.8	74	64
pr124	89	0.485805	64.8	102	0.505958	32.5	59	30
bier127	158	0.695374	67.8	242	0.702220	204.0	123	35
ch130	120	0.540726	67.0	147	0.554025	89.3	88	32
pr136	97	0.506539	95.0	152	0.520784	64.7	63	34
gr137	94	0.468528	115.0	129	0.478331	67.1	64	30
pr144	116	0.419821	110.3	133	0.442588	117.3	84	32
ch150	98	0.522792	113.8	154	0.538173	153.1	72	26
kroA150	94	0.511375	108.3	141	0.528899	120.2	74	20
kroB150	95	0.513359	121.2	110	0.538412	142.0	80	15

Table 10.12 Results for TSPLIB instances with profit type 1 and $m = 3$, part II

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
pr152	74	0.414320	110.8	114	0.431831	59.5	41	33
u159	103	0.478325	153.4	151	0.496434	137.3	68	35
rat195	112	0.476036	302.5	152	0.497755	381.8	93	19
d198	91	0.358539	259.6	163	0.382570	234.0	79	12
kroA200	102	0.508515	242.0	153	0.529265	383.7	92	10
kroB200	109	0.531025	225.2	168	0.546573	307.7	79	30
gr202	114	0.510636	323.1	160	0.522441	428.0	102	12
ts225	73	0.430202	342.8	116	0.460055	292.3	59	14
tsp225	85	0.449167	428.5	150	0.475739	541.3	71	14
pr226	76	0.474750	282.3	143	0.504124	164.9	53	23
gr229	116	0.614147	317.5	191	0.630407	922.4	104	12
gil262	89	0.493292	551.3	155	0.515675	811.5	68	21
pr264	109	0.483524	518.5	156	0.502934	545.6	103	6

(continued)

Table 10.12 (continued)

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
a280	108	0.467550	750.9	203	0.484065	911.2	96	12
pr299	95	0.467936	1141.7	201	0.487822	1245.8	70	25
lin318	121	0.508179	1298.5	212	0.525901	1847.1	108	13
rd400	158	0.547901	1540.2	261	0.560515	2064.3	127	31
fl417	118	0.535425	1411.1	285	0.547094	2199.9	100	18
gr431	192	0.668320	1866.0	299	0.673233	1738.2	101	91
pr439	118	0.537484	2465.5	171	0.548489	1169.2	73	45
pcb442	150	0.504819	2477.6	205	0.513141	1174.4	77	73
d493	92	0.404317	3600.9	92	0.404317	0.0	0	92
att532	116	0.560553	3600.8	116	0.560553	0.0	0	116

Table 10.13 Results for TSPLIB instances with profit type 1 and $m = 4$, part I

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
st70	76	0.478500	19.0	78	0.503710	7.6	49	27
eil76	99	0.524384	23.0	103	0.545408	20.0	86	13
pr76	73	0.492062	25.1	74	0.506035	8.0	47	26
gr96	70	0.512764	44.4	110	0.532737	21.6	49	21
rat99	89	0.451352	57.7	109	0.463070	23.3	52	37
kroA100	72	0.468140	41.9	103	0.495385	24.2	48	24
kroB100	82	0.541542	44.7	95	0.557907	21.4	59	23
kroC100	74	0.468278	47.9	91	0.488937	14.6	46	28
kroD100	83	0.479221	49.3	104	0.502895	33.7	67	16
kroE100	83	0.535865	47.7	107	0.552493	35.0	56	27
rd100	90	0.531184	49.3	114	0.551511	31.1	72	18
eil101	87	0.534391	44.1	89	0.551421	47.5	68	19
lin105	81	0.493667	60.7	105	0.510313	29.5	49	32
pr107	93	0.471421	66.8	121	0.476604	30.7	60	33
pr124	60	0.485479	83.0	108	0.509860	26.8	36	24
bier127	141	0.684695	83.0	216	0.693943	172.0	111	30
ch130	102	0.534579	88.4	140	0.549166	88.1	75	27
pr136	102	0.498581	120.0	127	0.517684	72.2	76	26
gr137	110	0.449393	149.0	133	0.465402	62.7	89	21
pr144	102	0.413376	139.8	126	0.430151	107.7	70	32
ch150	79	0.492262	155.1	113	0.508099	155.5	59	20
kroA150	97	0.485345	136.5	126	0.514431	166.7	85	12
kroB150	80	0.495800	160.5	121	0.512745	123.7	62	18

Table 10.14 Results for TSPLIB instances with profit type 1 and $m = 4$, part II

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
pr152	73	0.419057	144.9	145	0.440446	73.7	45	28
u159	104	0.457249	212.3	121	0.471292	138.0	84	20
rat195	90	0.466692	398.0	164	0.484506	386.9	70	20
d198	106	0.328646	327.7	181	0.341457	218.3	95	11
kroA200	95	0.485035	357.2	126	0.511350	475.8	85	10
kroB200	103	0.514041	294.6	158	0.533252	384.1	87	16
gr202	115	0.466900	388.7	172	0.479794	588.0	94	21
ts225	89	0.419131	492.4	126	0.446533	396.8	71	18
tsp225	81	0.452004	594.6	135	0.477020	428.3	70	11
r226	99	0.431950	411.8	160	0.452946	454.3	76	23
gr229	121	0.585081	462.6	192	0.606669	1002.3	113	8
gil262	89	0.455622	778.0	122	0.482332	967.6	80	9
pr264	100	0.465884	602.1	143	0.497300	609.6	92	8
a280	92	0.438428	1052.3	174	0.458690	1301.7	82	10
pr299	101	0.463710	1395.5	184	0.485422	1087.7	86	15
lin318	107	0.496853	1653.9	165	0.517104	1971.8	83	24
rd400	130	0.531509	2094.3	211	0.540525	1549.1	85	45
fl417	110	0.508544	1837.8	223	0.516112	1785.9	73	37
gr431	144	0.642946	2648.1	214	0.648455	986.4	81	63
pr439	117	0.554537	3526.0	148	0.555942	78.6	24	93
pcb442	117	0.489913	3600.5	117	0.489913	0.4	0	117
d493	94	0.373546	3600.0	94	0.373546	0.0	0	94
att532	102	0.537614	3600.7	102	0.537614	0.0	0	102

Table 10.15 Results for TSPLIB instances with profit type 3 and $m = 2$, part I

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
st70	105	0.475638	13.6	102	0.480869	4.6	45	60
eil76	100	0.519498	14.8	98	0.535149	17.3	68	32
pr76	76	0.498726	15.3	81	0.511959	9.1	48	28
gr96	107	0.496912	28.6	119	0.511531	30.8	71	36
rat99	113	0.438653	30.7	132	0.450850	29.2	74	39
kroA100	135	0.486347	28.6	137	0.498655	36.4	83	52
kroB100	110	0.500006	27.6	122	0.511038	18.4	72	38
kroC100	107	0.490497	30.4	115	0.499384	10.3	61	46
kroD100	92	0.491619	29.5	113	0.509572	17.2	68	24
kroE100	128	0.503505	29.3	146	0.514200	29.1	82	46
rd100	110	0.490617	31.3	107	0.508824	25.2	80	30
eil101	114	0.549285	27.3	133	0.561360	60.7	90	24
lin105	103	0.454122	32.3	127	0.469589	29.5	77	26
pr107	132	0.390145	34.4	167	0.399622	22.1	91	41

(continued)

Table 10.15 (continued)

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
pr124	120	0.481675	44.0	143	0.495953	19.8	89	31
bier127	144	0.567503	41.0	229	0.576649	171.4	114	30
ch130	127	0.512501	52.8	163	0.532513	110.7	98	29
pr136	119	0.482991	66.4	139	0.500636	66.8	87	32
gr137	116	0.427082	65.4	137	0.442061	52.9	80	36
pr144	119	0.412650	59.9	143	0.426932	41.3	85	34
ch150	108	0.488781	94.6	144	0.510137	126.9	85	23
kroA150	105	0.522259	82.4	118	0.539601	105.4	77	28
kroB150	98	0.507344	91.5	137	0.527565	82.3	78	20

Table 10.16 Results for TSPLIB instances with profit type 3 and $m = 2$, part II

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
pr152	128	0.398900	80.5	191	0.411503	38.3	73	55
u159	94	0.471524	90.5	149	0.490488	102.8	78	16
rat195	112	0.474745	155.5	155	0.489725	179.8	81	31
d198	118	0.369466	161.1	180	0.382628	243.5	97	21
kroA200	118	0.503387	188.3	162	0.521883	362.5	94	24
kroB200	106	0.516479	172.9	179	0.532347	301.3	83	23
gr202	138	0.517349	181.3	168	0.532859	617.9	116	22
ts225	104	0.454420	217.8	151	0.474769	303.9	76	28
tsp225	112	0.469847	231.8	191	0.489806	489.7	88	24
pr226	125	0.480364	187.4	190	0.499217	195.9	101	24
gr229	146	0.547607	181.0	224	0.562307	1266.8	134	12
gil262	115	0.469677	393.7	187	0.495426	876.7	106	9
pr264	116	0.469722	279.4	194	0.490142	175.8	98	18
a280	129	0.485949	357.4	227	0.502439	520.2	107	22
pr299	142	0.509108	483.0	253	0.523065	389.5	111	31
lin318	120	0.485770	552.2	222	0.502914	1077.6	96	24
rd400	160	0.532418	1180.3	283	0.544077	2437.6	117	43
fl417	140	0.502319	986.4	303	0.516286	1303.8	111	29
gr431	202	0.567471	989.3	326	0.573282	2682.8	88	114
pr439	135	0.546797	1075.7	248	0.564262	2660.4	113	22
pcb442	144	0.517142	1409.4	278	0.531853	2198.1	84	60
d493	127	0.448896	2917.0	187	0.453242	706.1	59	68
att532	145	0.495494	2638.2	213	0.502027	983.8	47	98

Table 10.17 Results for TSPLIB instances with profit type 3 and $m = 3$, part I

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
st70	94	0.445320	17.0	91	0.459187	3.9	56	38
eil76	94	0.494974	19.9	81	0.510206	6.5	51	43
pr76	65	0.454883	19.1	70	0.471705	4.1	37	28
gr96	88	0.445427	39.5	93	0.477085	37.2	74	14
rat99	104	0.414694	38.9	103	0.431438	17.6	71	33
kroA100	91	0.454487	38.6	96	0.475204	27.6	58	33
kroB100	91	0.481613	34.4	110	0.496598	24.9	63	28
kroC100	107	0.472680	40.3	114	0.483300	10.0	57	50
kroD100	101	0.475104	39.9	120	0.485873	11.9	66	35
kroE100	103	0.489792	37.0	127	0.502652	17.3	76	27
rd100	123	0.473408	40.6	122	0.490907	26.2	83	40
eil101	112	0.521074	35.3	113	0.539744	61.2	91	21
lin105	102	0.420155	45.8	112	0.438620	28.8	82	20
pr107	134	0.378232	41.8	153	0.386990	15.0	85	49
pr124	108	0.461442	57.9	128	0.485221	16.9	73	35
bier127	121	0.551837	65.0	160	0.565052	162.5	96	25
ch130	84	0.499054	73.8	119	0.519613	60.3	65	19
pr136	107	0.476238	86.4	146	0.491394	42.1	78	29
gr137	109	0.404468	93.6	137	0.420398	56.3	75	34
pr144	139	0.396397	75.8	152	0.409888	36.8	74	65
ch150	109	0.482422	121.3	136	0.499401	119.2	86	23
kroA150	99	0.506405	109.0	142	0.523202	131.9	84	15
kroB150	92	0.492464	116.4	118	0.508060	39.4	63	29

Table 10.18 Results for TSPLIB instances with profit type 3 and $m = 3$, part II

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
pr152	117	0.385507	101.7	150	0.397468	55.1	69	48
u159	122	0.451305	122.1	146	0.465126	78.0	86	36
rat195	127	0.472635	200.1	186	0.481950	132.6	88	39
d198	111	0.307502	205.9	141	0.327631	254.0	97	14
kroA200	137	0.500482	238.1	184	0.512678	183.4	106	31
kroB200	118	0.508705	213.5	172	0.526920	169.9	90	28
gr202	97	0.464535	268.6	146	0.484026	369.7	82	15
ts225	96	0.451576	282.1	158	0.471650	103.7	66	30
tsp225	101	0.454535	303.9	166	0.473465	232.8	82	19
pr226	136	0.460957	252.4	224	0.473276	292.9	87	49
gr229	119	0.506539	267.7	171	0.531340	483.0	112	7
gil262	112	0.451954	516.5	160	0.471737	719.9	98	14
pr264	108	0.476137	432.9	185	0.491492	297.0	87	21

(continued)

Table 10.18 (continued)

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
a280	115	0.473043	490.8	169	0.487371	871.7	86	29
pr299	129	0.472274	711.2	198	0.493908	1302.7	111	18
lin318	125	0.470040	780.5	215	0.485324	766.6	101	24
rd400	138	0.508452	1748.0	210	0.523157	1894.1	102	36
fl417	121	0.515440	1355.1	281	0.529273	1718.4	105	16
gr431	132	0.522532	1836.4	218	0.527042	1791.1	61	71
pr439	128	0.547026	1550.7	265	0.560636	1128.1	115	13
pcb442	92	0.492013	2085.5	198	0.503944	1520.8	52	40
d493	115	0.399872	3478.6	150	0.401332	135.9	28	87
att532	101	0.446685	3602.0	101	0.446685	0.0	0	101

Table 10.19 Results for TSPLIB instances with profit type 3 and $m = 4$, part I

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
st70	76	0.430686	20.6	89	0.449664	5.5	35	41
eil76	86	0.466485	24.0	82	0.480635	12.0	57	29
pr76	113	0.409441	23.1	86	0.431792	9.7	73	40
gr96	102	0.444316	43.7	96	0.460763	22.6	64	38
rat99	96	0.407039	47.2	101	0.418169	23.8	65	31
kroA100	110	0.467525	44.8	130	0.472698	16.3	41	69
kroB100	100	0.474129	43.2	107	0.488517	16.8	70	30
kroC100	89	0.444637	44.3	91	0.459940	16.5	53	36
kroD100	98	0.456582	47.0	107	0.472653	15.1	72	26
kroE100	104	0.480908	46.2	114	0.491166	19.0	57	47
rd100	114	0.464951	47.5	128	0.474138	21.9	60	54
eil101	84	0.506713	43.4	107	0.523025	29.3	59	25
lin105	116	0.419691	56.4	118	0.434316	26.2	83	33
pr107	91	0.371506	45.1	124	0.381704	20.1	63	28
pr124	122	0.471328	67.9	130	0.477463	31.4	50	72
bier127	111	0.542659	83.7	164	0.556650	144.0	92	19
ch130	111	0.495598	91.7	123	0.507498	60.2	71	40
pr136	106	0.451855	103.9	137	0.470404	59.2	95	11
gr137	108	0.381051	110.4	128	0.396918	80.4	70	38
pr144	138	0.397850	86.1	161	0.407388	36.8	80	58
ch150	93	0.478111	140.8	123	0.497006	87.3	68	25
kroA150	101	0.494357	126.9	126	0.505698	29.6	59	42
kroB150	94	0.468010	148.2	109	0.490788	75.8	80	14

Table 10.20 Results for TSPLIB instances with profit type 3 and $m = 4$, part II

Instance	NSGA-II			PR			N_{dom}	N_{eq}
	N_1	Hyp	Time (s)	N_2	Hyp	Time (s)		
pr152	88	0.374362	123.9	130	0.391891	34.6	64	24
u159	128	0.435006	155.4	146	0.446770	82.5	100	28
rat195	111	0.457019	263.5	159	0.470328	226.1	79	32
d198	108	0.286206	252.1	164	0.299729	188.6	81	27
kroA200	122	0.494745	300.0	172	0.509883	162.8	95	27
kroB200	116	0.495731	266.4	157	0.508347	199.0	89	27
gr202	108	0.421093	328.5	145	0.443903	549.2	98	10
ts225	92	0.453527	375.9	139	0.470380	157.7	69	23
tsp225	119	0.445973	391.6	198	0.461090	275.0	81	38
pr226	111	0.438803	342.0	178	0.457603	129.4	71	40
gr229	127	0.498635	335.6	162	0.516818	439.6	112	15
gil262	133	0.458874	644.7	189	0.479200	474.1	116	17
pr264	109	0.470985	476.0	194	0.484733	296.1	84	25
a280	132	0.470783	636.9	201	0.481105	500.1	92	40
pr299	99	0.465195	956.5	229	0.479157	528.0	77	22
lin318	118	0.465885	962.6	209	0.481869	1107.4	98	20
rd400	97	0.516764	2396.7	186	0.525216	1216.1	52	45
fl417	99	0.494772	1974.6	221	0.506046	1687.1	73	26
gr431	128	0.488365	2616.8	205	0.492872	1016.2	42	86
pr439	162	0.546597	2085.4	259	0.556669	1550.6	111	51
pcb442	91	0.489556	2729.5	150	0.497674	919.1	34	57
d493	114	0.368130	3600.0	114	0.368130	0.1	0	114
att532	125	0.456692	3601.4	125	0.456692	0.0	0	125

10.6 Conclusion

We have presented a first study of the bi-objective multiple TSP with profits, a generalization of the TSPP. The solution approach was based on NSGA-II algorithm featured with a simple solution encoding represented by a sequence of customers without trip delimiters. A decoding procedure was developed for obtaining the real representation of solutions. This split procedure enables to retrieve the optimal solution with respect to the encoding in polynomial time.

The Pareto front approximation is further improved with path relinking. The implemented PR algorithm explores trajectories connecting two solutions which are adjacent in the objective space. Computational results prove the ability of PR to improve significantly the Pareto front approximation found by NSGA-II. This concept has been seldom concerned in the multi-objective optimization research and it seems that it can be a beneficial complement to many of the Pareto front approximation algorithms.

Further research might be conducted toward the reduction of computing time required by both components of the proposed algorithm. The solution approach can be further developed in order to tackle more general problems, for example, more objective functions or additional constraints.

References

Archetti C, Hertz A, Speranza MG (2007) Metaheuristics for the team orienteering problem. J Heur 13(1):49–76

Basseur M, Seynhaeve F, Talbi EG (2005) Path relinking in pareto multi-objective genetic algorithms. In: Coello Coello CA, Hernndez Aguirre A, Zitzler E (eds) Evolutionary multi-criterion optimization. Thrid international conference, EMO 2005. Lecture notes in computer science, vol 3410. Springer Mexico, Guanajuato, pp 120–134.

Beausoleil R, Baldoquin G, Montejo R (2008) Multi-start and path relinking methods to deal with multi-objective knapsack problems. Ann Oper Res 157:105–133

Bérubé JF, Gendreau M, Potvin JY (2009a) An exact ε-constraint method for bi-objective combinatorial optimization problems: application to the traveling salesman problem with profits. Eur J Oper Res 194(1):39–50

Bérubé JF, Gendreau M, Potvin JY (2009b) A branch-and-cut algorithm for the undirected prize collecting traveling salesman problem. Networks 54:56–67

Bouly H, Dang DC, Moukrim A (2010) A memetic algorithm for the team orienteering problem. 4OR: Q J Oper Res 8(1):49–70

Boussier S, Feillet D, Gendreau M (2007) An exact algorithm for team orienteering problems. 4OR: Q J Oper Res 5(3):211–230

Chao IM, Golden BL, Wasil EA (1996) The team orienteering problem. Eur J Oper Res 88(3):464–474

Dantzig GB (1957) Discrete-variable extremum problems. Oper Res 5(2):266–288

Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans Evo Comput 6(2):182–197

Feillet D, Dejax P, Gendreau M (2005) Traveling salesman problems with profits. Transp Sci 39(2):188–205

Glover F, Laguna M, Marti R (2000) Fundamentals of scatter search and path relinking. Cont Cyber 39:653–684

Jaszkiewicz A, Zielniewicz P (2009) Pareto memetic algorithm with path relinking for bi-objective traveling salesperson problem. Eur J Oper Res 193(3):885–890

Jozefowiez N, Glover F, Laguna M (2008) Multi-objective meta-heuristics for the traveling salesman problem with profits. J Math Mod Alg 7(2):177–195

Ke L, Archetti C, Feng Z (2008) Ants can solve the team orienteering problem. Comput Ind Eng 54(3):648–665

Keller CP, Goodchild M (1988) The multi-objective vending problem: a generalization of the travelling salesman problem. Environ Plan B: Plan Des 15:447–460

Konak A, Coit DW, Smith AE (2006) Multi-objective optimization using genetic algorithms: a tutorial. Reliab Eng Syst Saf 91(9):992–1007

Labadi N, Prins C, Reghioui M (2008) Grasp with path relinking for the capacitated arc routing problem with time windows. In Fink A and Rothlauf F (eds) Advances in computational intelligence in transport, logistics, and supply chain management. Studies in computational intelligence, vol 144. Springer, Berlin, pp 111–135

Labadie N, Melechovsky J, Wolfler Calvo R (2010) An effective hybrid evolutionary local search for orienteering and team orienteering problem with time windows. In: Schaefer R et al (eds) Lecture Notes in Computer Science, vol 6239. Springer Berlin, Heidelberg, pp 219–228

Labadie N, Melechovsky J, Wolfler Calvo R (2011) Hybridized evolutionary local search algorithm for the team orienteering problem with time windows. J Heur 17(6):729–753

Labadie N, Mansini R, Melechovský J, Wolfler-Calvo R (2012) The team orienteering problem with time windows: an LP-based granular variable neighborhood search. Eur J Oper Res 220(1):15–27

Martí R, Laguna M, Campos V (2005) Scatter search vs. genetic algorithms. In: Ramesh Sharda, Stefan Vo, Csar Rego, Bahram Alidaee, Ramesh Sharda, and Stefan Voss (eds) Metaheuristic optimization via memory and evolution. Operations Research/Computer Science Interfaces Series, vol 30. Springer, New York, pp 263–282

Martí R, Campos V, Resende MGC, Duarte A (2011) Multi-objective grasp with path-relinking. Technical report, AT and T Labs Research

Montemanni R, Gambardella LM (2009) An ant colony system for team orienteering problem with time windows. Found Comput Dec Sci 34(4):287–306

Prins C (2004) A simple and effective evolutionary algorithm for the vehicle routing problem. Comput Oper Res 31(12):1985–2002

Reinelt G (1991) TSPLIB—A traveling salesman problem library. ORSA J Comput 3(4):376–384

Riera-Ledesma J, Salazar-González JJ (2005) The biobjective travelling purchaser problem. Eur J Oper Res 160(3):599–613

Schilde M, Doerner KF, Hartl RF, Kiechle G (2009) Metaheuristics for the bi-objective orienteering problem. Swarm Intell 3(3):179–201

Tang H, Miller-Hooks E (2005) A tabu search heuristic for the team orienteering problem. Comput Oper Res 32(6):1379–1407

Tricoire F, Romauch M, Doerner KF, Hartl RF (2010) Heuristics for the multi-period orienteering problem with multiple time windows. Comput Oper Res 37(2):351–367

Tsiligirides T (1984) Heuristic methods applied to orienteering. J Oper Res Soc 35(9):797–809

Vansteenwegen P, Souffriau W, Berghe GV, Oudheusden DV (2009a) In: Metaheuristics in the service industry. Lecture notes in economics and mathematical systems, chapter metaheuristics for tourist trip planning, vol 624. Springer Berlin, Heidelberg, pp 15–31

Vansteenwegen P, Souffriau W, Berghe GV, Oudheusden DV (2009b) Iterated local search for the team orienteering problem with time windows. Comput Oper Res 36(12):3281–3290

Vansteenwegen P, Souffriau W, Oudheusden DV (2011) The orienteering problem: a survey. Eur J Oper Res 209:1–10

Wang X, Golden BL, Wasil EA (2008) Using a genetic algorithm to solve the generalized orienteering problem. In: Golden B, Raghavan S, Wasil EA (eds) The vehicle routing problem: latest advances and new challenges. Springer, pp 263–274

Zitzler E (1999) Evolutionary algorithms for multi-objective optimization: methods and applications. Ph.D. thesis, Swiss Federal Institute of Technology, Zurich

Part II
Game Theory Applications

Chapter 11
A Hybrid Simulation-based Duopoly Game Framework for Analysis of Supply Chain and Marketing Activities

D. Xu, C. Meng, Q. Zhang, P. Bhardwaj and Y. J. Son

Abstract A hybrid simulation-based framework involving system dynamics (SD) and agent-based simulation (ABS) is proposed to address duopoly game considering multiple strategic decision variables and rich payoff, which cannot be addressed by traditional approaches involving closed-form equations. While SD models are used to represent integrated production, logistics, and pricing determination activities of duopoly companies, ABS is used to mimic enhanced consumer purchasing behavior considering advertisement, promotion effect, and acquaintance recommendation in the consumer social network. The payoff function of the duopoly companies is assumed to be the net profit based on the total revenue and various cost items such as raw material, production, transportation, inventory and backorder. A unique procedure is proposed to solve and analyze the proposed simulation-based game, where the procedural components include strategy refinement, data sampling, gaming solving, and performance evaluation. First, design of experiment (DOE) and estimated conformational value of information (ECVI) techniques are employed for strategy refinement and data sampling, respectively. Game solving then focuses on pure strategy equilibriums, and performance evaluation addresses game stability, equilibrium strictness, and robustness. A hypothetical case scenario involving soft-drink duopoly on Coke and Pepsi is considered to illustrate and demonstrate the proposed approach. Final results include *p*-values of statistical tests, confidence intervals, and simulation steady state analysis for different pure equilibriums.

Keywords Simulation-based game · System dynamics · Agent-based simulation · Equilibrium · Soft-drink duopoly

D. Xu · C. Meng · Q. Zhang · Y. J. Son (✉)
Systems and Industrial Engineering, The University of Arizona, Tucson, AZ, USA
e-mail: son@sie.arizona.edu

P. Bhardwaj
Operational Decision Support Technology, Intel Corporation, Tempe, AZ, USA

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_11,

11.1 Introduction

Duopoly games have been extensively studied in the modern history of economics, where the market is primarily dominated by two major companies and they make fully rational decisions to reach the goals (e.g. maximize payoff). While the most widely used approaches to solve the duopoly game are based on Cournot model (Cournot 2012) and Bertrand model (Bertrand 1883), several major limitations of those models are that:

- The *payoff function* of each company is highly aggregated by closed-form mathematical equations;
- Only single or limited decision variables (e.g. production quantity, product price) are considered for mathematical tractability;
- No randomness involved in the payoff formulation.

In real practice, however, competing companies have to make and update decisions periodically on various areas such as production, logistics and price across the entire supply chain based on dynamically changing market conditions, and these decisions interact with one another to achieve a high profit. Hence, a comprehensive modeling technique is desired to mimic the realistic processes in multiple areas mentioned above, so as to provide a highly accurate payoff as well as to enable analysis of the trade-offs among different strategies.

In this chapter, a hybrid simulation-based framework is proposed to address duopoly game under the scenario of product adoption process considering multiple decision variables and detailed payoffs. In the proposed hybrid simulation framework,

- *system dynamics* (SD) models are used for simulating the activities of duopoly companies on production, logistics, and price determination;
- *agent-based simulation* (ABS) is used for modeling consumer purchasing behaviors at the market side.

Figure 11.1 outlines the major components in an exemplary supply chain and consumer market. In the SD model, an integrated production-logistics model considering the material transformations and flows from suppliers to final customers is constructed for each duopoly company. The price determination process, which is also modeled in the SD simulation, represents how each company determines the product price and adjusts it over time due to the impacts of production and logistics. To this end, an enhanced consumer motivation function is developed based on various factors such as the effect of advertisement, the effect of promotion, the influences of customer acquaintance recommendation, and the price sensitivity in the consumer social network. The consumer motivation function is then incorporated into the ABS for mimicking the consumer purchasing behaviors, which is tightly coupled with the SD model for the duopoly companies.

Considering the *game strategy* for duopoly games, emphasis has been put in the following strategic areas (Min and Zhou 2002; Hong et al. 2008; Song and Jing 2010) including production strategy (e.g. labor, raw material availability), logistics

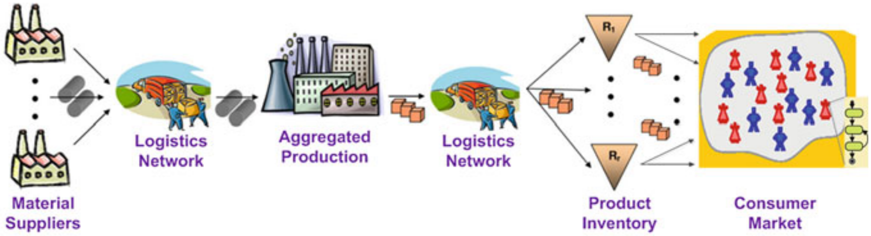


Fig. 11.1 Exemplary supply chain and consumer market

strategy (e.g. lead time, inventory coverage control), and marketing strategy (e.g. price determination, advertising, promotion). The strategic areas in the literature are coupled with the simulation model, so that any strategy changes can be reflected in simulation variables/parameters. In the game theory literature, each of the above strategic areas involves different decisions that are referred to as strategies. The payoff function of each dominated company is defined in terms of net profit, which is the difference between the revenue and various cost items such as production, logistics, transportation, and backorder. In the proposed work of this chapter, the objective for each duopoly company is assumed to maximize the net profit via the coordination of all the considered strategies.

In games involving a large number of strategies and data samples, conducting experiments including all the strategic decisions is computationally costly. In order to solve and analyze the *simulation-based game* in this work considering limited computational resources, a novel procedure is proposed, where the procedural components include strategy refinement, data sampling, game solving, and performance evaluation. First, design of experiments (DOE) technique used for strategy refinement and ECVI technique used for data sampling are integrated for exploring the strategy space in the empirical game setting. Then, game solving for pure strategy equilibrium is applied to generate game equilibrium results, and performance evaluation approach is employed to assess various output criteria (e.g. equilibrium quality, stability, strictness and robustness). In the experiment section, a case with soft-drink duopoly game is considered to illustrate and demonstrate the framework.

Figure 11.2 depicts major components of the framework in this chapter:

- A hybrid simulation testbed of duopoly game with its profile set as inputs and payoff matrix as outputs (the upper part of Fig. 11.2);
- A game solving and analysis (GSA) procedure including strategy refinement, data sampling, game solving and performance evaluation (the lower part of Fig. 11.2).

The major contributions of this chapter are summarized as follows:

1. A novel simulation-based *empirical game* platform is proposed, which overcomes the major drawbacks of closed-form mathematical equations in terms of modeling comprehensiveness;

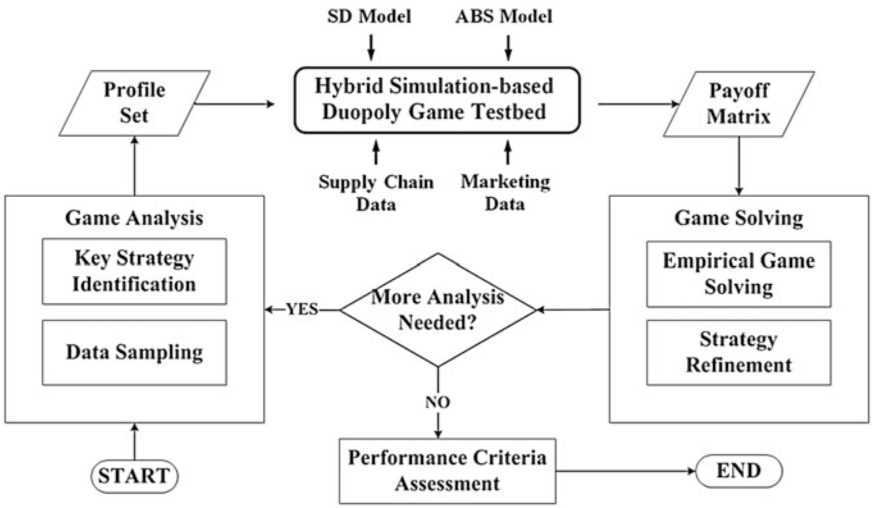


Fig. 11.2 Proposed hybrid simulation framework with the GSA procedure

2. A novel simulation-based *game solving and analysis* (GSA) procedure is proposed, which covers major topics in the field of game theory such as strategy refinement, data sampling, game solving, and performance evaluation.

In fact, the proposed simulation platform allows for accurate representation of the real world scenario, and it targets to address such game problem involving large strategy space and detailed/rich payoff function. Besides, the proposed platform is generic so that it can be re-used and further enhanced based on user requirements. The proposed GSA procedure is platform independent so that it can also be applied to resolve other similar simulation-based games.

The rest of this chapter is organized as follows. In Sect. 11.2, the literature works related to the proposed simulation-based game platform and the GSA procedure are summarized. In Sect. 11.3, the details of different modeling aspects (e.g. production, logistics, and marketing) that constitute the simulation-based game platform are provided, followed by the discussions of game strategies and payoff function. Section 11.4 discusses the motivation, objective of GSA, as well as its detailed procedure including strategy refinement, data sampling, game solving, and performance assessment. In Sect. 11.5, experiments are conducted and corresponding results are presented. Finally, conclusions and future directions are discussed in Sect. 11.6.

11.2 Background and Literature Survey

The game theoretic approach has been applied in the literature to address strategic decision problems in supply chain and marketing activities, where the studies mainly focused on the relationships between stakeholders within the supply chain system. For manufacturing strategy, Zhang and Huang (2010) investigated platform commonality and modularity strategies in a supply chain consisting of a single manufacturer and multiple cooperative suppliers. They derived the optimal ordering and pricing decisions for the two-moves dynamic game according to Nash's bargaining model, and developed an iterative algorithm to find the subgame perfect equilibrium. They found that a supply chain with cooperative suppliers is more effective by using the lot-for-lot policy and more competitive by accommodating higher product variety. For logistics/inventory control strategy, Yu et al. (2006) studied Stackelberg game in a Vendor Managed Inventory (VMI) supply chain consisting one manufacturer as the leader and heterogonous retailers as followers. The research proposed a 5-step algorithm to reach the Stackelberg equilibrium and demonstrated (1) the significant influence of market-related parameters on manufacturer's and retailers' profit, (2) higher inventory cost does not necessarily lead to lowing retailers' profit and (3) game equilibrium benefits the manufacturer. The pricing and marketing strategies have been studied in an integrated manner in some literature works. Parlar and Wang (1994) studied the discounting strategy in a game involving one supplier with multiple homogeneous customers. They demonstrated that both seller and buyers can improve their own profit by using a proper discounting strategy. A similar game was also studied by Wang and Wu (2000). The difference was that the customers in this study were heterogeneous, and a price policy was proposed, where seller offers price discount based on the percentage increase from a buyer's quantity before discount. The proposed policy was demonstrated to provide benefits for venders compared with the one based on buyer's unit increase in order quantity. Esmaeili et al. (2009) proposed seller-buyer supply chain models considering pricing and marketing strategic decision variables such as price charged by seller to buyer, lot size, buyer's selling price, and marketing expenditure. Both cooperative and non-cooperative relationships between the seller and buyer were modeled assuming seller-stackelberg and buyer-stackelberg, respectively. The experiment results showed both optimal selling price and marketing expenditure were smaller in the cooperative game. While these works have provided guidance for addressing strategic decision making problems via a game theoretic approach, they faced limitations in efficiently obtaining accurate payoffs for a large strategy space under realistic case scenarios (e.g. duopoly company competition).

Most recently, simulation-based games have been employed to analyze complex interactions of players in the areas of supply chain (Collins et al. 2004), combat (Poropudas and Virtanen 2010), financial market (Mockus 2010), subcontractor selection (Unsal and Taylor 2011) and pedestrian behaviors (Asano et al. 2010). An advantage of this approach is that simulation is capable of modeling the detailed players' behaviors, their interactions as well as the external environment impacts. Hence, results from simulation are comprehensive and can be used for detailed analysis. To the best of our knowledge, although simulation-based game has been used for solving coordination problem within specific supply chain, a formal framework for solving integrated supply chain and its market competition game is needed in literature. Next several paragraphs mainly survey the past research works that have formed a basis in this chapter in two aspects:

- SD and ABS modeling on supply chain and marketing activities, respectively;
- Approaches for empirical game analysis.

Concerning the simulation model for integrating the supply chain operations and marketing activities, different researchers have developed scenarios with distinct settings according to their own conveniences. To unify them under a coherent framework, the SD model in our work employs typical scenarios available in Sterman (2000) that involve labor utilization, raw material logistics, production process, and final production inventory control. However, necessary modifications have been made due to the duopoly game setting, and ABS integration for consumer purchasing behavior (see Sect. 11.3 for details). The consumer purchasing motivation and decision can be influenced by three factors (Kotler and Keller 2007):

- Personal (e.g. price sensitivity and quality sensitivity);
- Social (e.g. adoption from word of mouth, follower tendency);
- Psychological (e.g. perception and susceptibility to advertisement).

ABS can not only explore how and why consumers made the decision of purchasing certain products (North et al. 2010), but also evaluate the overall system performance without sacrificing enough details on interdependency among company marketing behaviors. Previous researchers (Jager et al. 1999; Adjali et al. 2005; Yoshida et al. 2007) have dealt with personal, social and psychological factors involving ABS technique. In this chapter, based on Zhang and Zhang (2007), an enhanced motivation function is proposed to incorporate the effects of advertisement, promotion from company, the influences of customer acquaintance recommendation, and price sensitivity in the consumer market. The consumer behavior modeled in ABS is coupled with the supply chain model to generate the market share and actual demand over time.

Previous literature works related to the simulation-based game analysis of this chapter are summarized in the following two paragraphs. A seminal research work in empirical game analysis is Wellman (2006), who decomposed the empirical game-theoretic analysis into three basic steps:

1. Parameterize strategy space, which means to generate a set of candidate strategies from all available ones that are computationally intensive and costly ineffective to evaluate;
2. Estimate the empirical game, which is aimed to construct empirical payoff matrix via simulation for the simplified game with the attention on the candidate strategies;
3. Analyze (solve) the empirical game, and assess the solution quality with respect to the original game with full strategy sets.

For parameterizing strategy space, several baseline approaches are available in Wellman (2006) such as truthful revelation, myopic best response and game tree search. These methods have been applied in auction game (Reeves 2005) and multi-player chess game (Kiekintveld et al. 2006). For estimating the empirical game, two approaches exist in the literature, including direct estimation and regression. The first approach treats the observations as direct evidence for the payoffs of each player's strategy profile, while the idea of second method is to apply regression to fit an estimated payoff function over the entire profile space given the available data (Vorobeychik et al. 2007). The goal of analyzing the empirical game is to find the pure and mixed strategy equilibrium firstly, and then apply appropriate methods (e.g. statistical bounds) to gain insights into the original full game. Degree of game-theoretic stability is usually used to provide an ε-Nash equilibrium under this case.

Similar to our strategy refinement problem addressed in this chapter, Jordan et al. (2008) studied the profile selection problem with the objective of saving the computational costs for the promising equilibrium candidates. The authors studied different algorithms applicable to two different models: TABU best-response search (Sureka and Wurman 2005) and minimum regret-first search (MRFS) for revealed-payoff; expected value of information (EVI) (Walsh et al. 2003) and proposed information gain (IG) approach for noisy-payoff models. Later on, Jordan et al. (2010) solved a special case of the profile selection problem to determine an optimal simulation sequence of strategy sets. The paper also clarified the differences between the profile selection problem (Jordan et al. 2008) and strategy exploration problem. Then, different exploration policies including random policy (RND), improving deviation only policy (DEV), best response policy (BR), softmax policy (ST) were discussed, followed by the experiments to compare their performances under different scenarios. For the sampling approach, Walsh et al. (2003) referred to the large/infinite number of strategies in the populated strategy space as heuristic strategies, and proposed two information theoretic approaches (i.e. EVI and ECVI approaches) to compute the additional sampling number for each experimental step. The paper demonstrated that ECVI approach converged faster than EVI given the same number of samples, and they both outperformed the uniform sampling approach. As pointed out in these literature works, when dealing with a large game strategy space, strategy exploration/refinement and data sampling are always the dominant costs for solving and analyzing the game, which constitute the major motivation for the development of the proposed GSA procedure in Sect. 11.4.

11.3 Hybrid Simulation-based Testbed for Duopoly Game Modeling

In this section, two major functional components constitute the simulation testbed: supply chain and marketing. The supply chain operations are modeled in SD, and marketing activities with its impact to the consumer behavior are modeled in

ABS. The supplying process at the upstream is responsible for providing raw materials to the manufacturer. Production at the manufacturer begins when both raw material and labor are available. Inventories are kept along the supply chain to satisfy the customer orders at the downstream, and backorder is considered when the demand can't be fulfilled. The product price is also determined in the SD model, and it is impacted by the competitive product in the market, and production-logistics activities of its own company. Consumer purchasing behaviors are represented in the ABS model, which are highly related to the companies' market share and profit. All these modeling details are presented in the rest of this section.

11.3.1 System Dynamics for Modeling Production-Logistics Activities Under Duopoly Game Environment

Figures 11.3 and 11.4 are the snapshots for the production and logistics modules in the SD model, respectively, where equations from (11.1) to (11.21) represent underlying mathematical models and Table 11.1 provides nomenclatures for variables and parameters used in those equations. The concepts behind the SD model developed in this chapter are based on Sterman (2000) and Venkateswaran and Son (2007), with the enhancements and customizations made for our study. The major customizations/enhancements include:

- Duopoly game setting for our scenario;
- Interaction with the marketing module in ABS model;
- Incorporation of historical values via exponential moving average for adjusted production, inventory, labor, and vacancy;
- Incorporation of variations in demand, production, inventory, labor availability.

The entire production process has been aggregated into one stock in the SD model [see Eq. (11.1)]. One assumption made when constructing these equations is that we treat the time as discrete variable, while in the SD model the corresponding variables change continuously. The adjusted WIP and production amounts are calculated via *exponential moving average* (smoothing) as shown in Eqs. (11.2) and (11.3). As it is an order-driven inventory control and production system, the desired amount of WIP is calculated by multiplying the total of adjusted production amount and customer order rate with the manufacturing cycle time plus the variations (see Eq. (11.4)); the desired amount of production begin rate is calculated by summing up the adjusted WIP amount, adjusted production amount, customer order rate and the variations [see Eq. (11.5)].

In the ideal case, the actual production begin rate is equal to the desired production begin rate; however, it is always constrained by two other factors: *workforce availability* and *raw material availability* [see Eq. (11.6)]. The availability of raw material is determined by the upstream supplier, of which the modeling is analogous to the logistics module of the finished goods (discussed

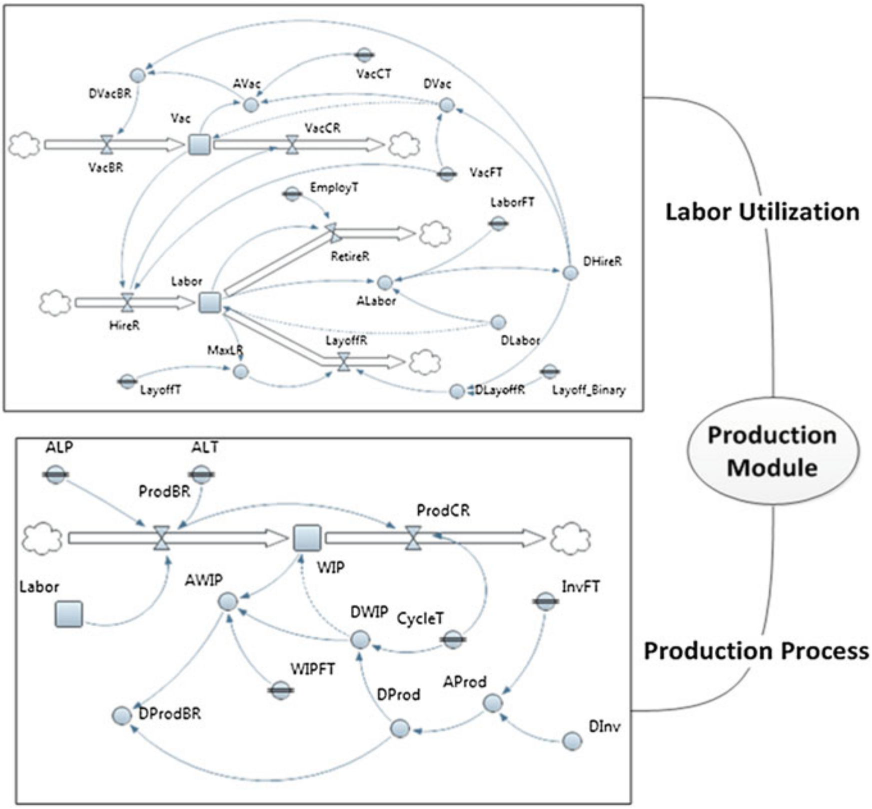


Fig. 11.3 Production module in the simulation-based game testbed (customized from Sterman (2000))

later in this section). The labor changing process (e.g. vacancy creation and fulfillment) will be discussed in the next paragraph. The actual production begin rate equals to the minimal one (bottleneck) of the workforce, raw material amount, and desired production begin rate. The production cost is tightly related to the product price, which will be discussed later in this section.

One factor that influences the production plan is the labor availability. The labor is represented in one stock, and the labor vacancy rate is captured in another stock. The equations for calculating these two stock values are shown in Eqs. (11.7) and (11.8). Hiring rate, retiring rate, and layoff rate are explicitly modeled in the SD model via Eqs. (11.9)–(11.11), respectively. These three rates are the major variables for deciding the labor availability, and a variable called vacancy begin rate will be increased if the SD model desires more labor. The vacancy begin rate is computed by the adjusted amounts of labor and vacancy in total [see Eq. (11.12)]. And the adjusted amounts of labor and vacancy are calculated via exponential moving average (smoothing) in Eqs. (11.13) and (11.14). Finally, the

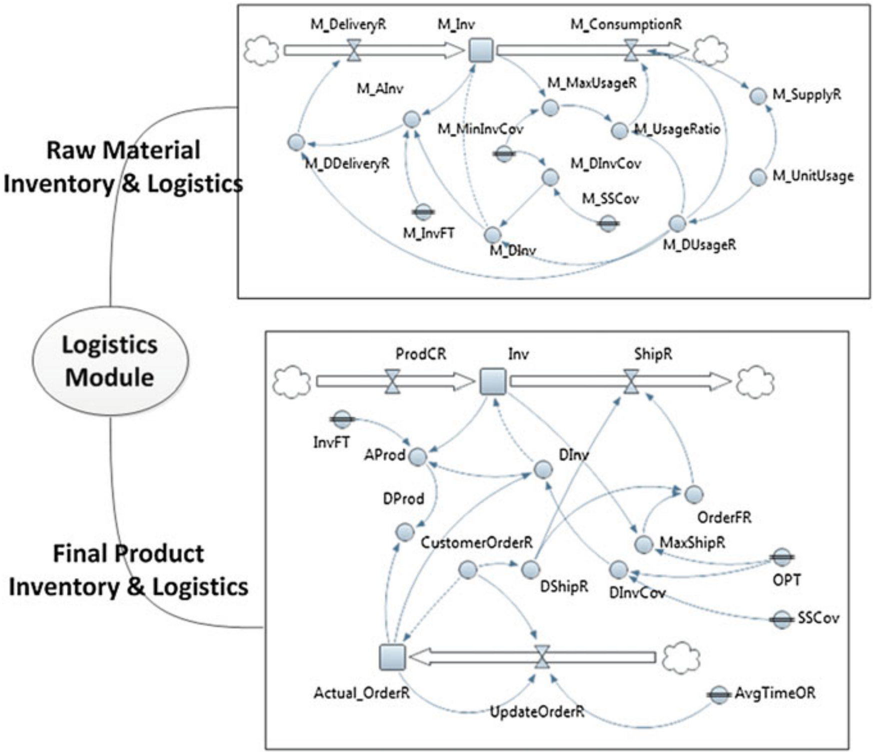


Fig. 11.4 Logistics module in the simulation-based game testbed (customized from Sterman (2000))

desired amounts of labor and vacancy are calculated in Eqs. (11.15) and (11.16), which are similar to the calculations of desired production and inventory. The decision variables considered in the production module are vacancy creation time (*VacCT*), average time for layoff labors (*LayoffT*), labor fulfillment time (*LaborFT*), and WIP fulfillment time (*WIPFT*).

The logistics part of the SD model is constituted with transportation and inventory control components. As the transportation lead time is simply used, it is translated into inventory fulfillment time for the ease of analysis and the following discussions in this paragraph focus only on the inventory part. Similar to the production process, one stock is used to aggregate the entire product inventory, and it is calculated in Eq. (11.17). A retailer maintains an inventory of finished goods, and fills orders as they arrive from customers. The desired shipment rate is set to be equal to the customer demand, while the actual shipment rate depends on the inventory level of the supply chain system. The customer order rate is calculated in Eq. (11.18), in which the market share and effects of advertisement and promotion are explicitly considered. The un-fulfilled amount of goods will be accounted into backlog inventory, and is calculated for the backorder cost. The

Table 11.1 Nomenclature for system dynamics model

Notation	Explanation	Notation	Explanation
ProdBR	Production begin rate	*SSCov*	Safety stock coverage
Labor	Labor amount	*OrderR*	Order rate
ALT	Average labor working time per time period	*HireR*	Labor hiring rate
ALP	Average labor productivity per time period	*RetireR*	Labor retire rate
DProdBR	Desired production begin rate	*LayoffR*	Labor layoff rate
AWIP	Adjustment amount for work-in-process (WIP)	*VacBR*	Vacancy begin rate
DProd	Desired production	*DHireR*	Desired labor hiring rate
DWIP	Desired amount of work-in-process	*LaborFT*	Labor fulfillment time
WIP	Amount of work-in-process product	*DVac*	Desired amount of vacancy
WIPFT	Fulfillment time for work-in-process product	*AVac*	Adjustment amount for vacancy
CycleT	Manufacturing cycle time	*MSR*	Raw material supplying rate
ProdCR	Production complete rate	*Vac*	Labor vacancies
AProd	Adjustment amount for production	*VacFT*	Average time to fill vacancies
DInv	Desired inventory level	*EmployT*	Average time of employment
Inv	Actual inventory level	*MaxLR*	Maximum layoff rate
InvFT	Fulfillment time for inventory	*VacCR*	Vacancy closure rate
OPT	Order processing time	*DLabor*	Desired labor
InvCov	Inventory coverage	*ALabor*	Adjustment number of labor
ShipR	Shipment rate	*LayoffT*	Average time for layoff labors
MS	Market share	*TOR*	Total order rate
MaxInvCov	Capacity of inventory coverage	$c^{(P)}$	Unit production cost
Price	Product price	*MP*	Market expected price
$PSens^{(C)}$	Price sensitivity to cost	$PSens^{(I)}$	Price sensitivity to inventory coverage
$F^{(C)}$	Effect of inventory coverage on price	$F^{(I)}$	Effect of cost on price
PriceCR	Price changing rate	*MPFT*	Fulfillment time of market expected price
$\sigma^{(W)}$	Variations for desired WIP	$\lambda^{(W)}$	Exponential smoothing factor for adjusted WIP
$\sigma^{(P)}$	Variations for desired production begin rate	$\lambda^{(P)}$	Exponential smoothing factor for adjusted production
$\sigma^{(O)}$	Variations for order rate	$\lambda^{(L)}$	Exponential smoothing factor for adjusted labor
$\sigma^{(I)}$	Variations for desired inventory level	$\lambda^{(V)}$	Exponential smoothing factor for adjusted vacancy

Subscripts i and t are omitted. i is player index (i = A, B), t represents simulation time order fulfillment ratio is then calculated based on the percentage of order being fulfilled, which is used to decide the actual shipment rate in Eq. (11.19). Equation (11.20) calculates the desired inventory level, which equals to the sum of minimal order processing time and safety level of stock, multiplied by the customer order rate. The variations are also included in Eq. (11.20). The inventory coverage represents the time duration that the current inventory level under the current shipment rate can cover the customer order, and is a superior measure of both goods holding cost for the supply chain members and the capability of buyers to receive reliable and timely deliveries. This variable is calculated in Eq. (11.21), and also used to decide the inventory effects to the product price. The decision variables considered in the logistics module are inventory fulfillment time (*InvFT*), raw material transportation lead time (*M_LT*), product safety stock coverage (*SSCov*), and raw material inventory coverage (*M_InvCov*).

$$WIP_{i,t} = \int_0^t (ProdBR_{i,s} - ProdCR_{i,s})ds \tag{11.1}$$

$$AWIP_{i,t} = \lambda^{(W)}(DWIP_{i,t} - WIP_{i,t-1})/WIPFT_i + (1 - \lambda^{(W)})AWIP_{i,t-1} \tag{11.2}$$

$$AProd_{i,t} = \lambda^{(P)}(DInv_{i,t} - Inv_{i,t-1})/InvFT_i + (1 - \lambda^{(P)})AProd_{i,t-1} \tag{11.3}$$

$$DWIP_{i,t} = (AProd_{i,t} + OrderR_{i,t}) \times E(CycleT_i) + \sigma^{(W)} \tag{11.4}$$

$$DProdBR_{i,t} = AWIP_{i,t} + AProd_{i,t} + OrderR_{i,t} + \sigma^{(P)} \tag{11.5}$$

$$ProdBR_{i,t} = \max(0, \min(Labor_{i,t} \times ALP_i \times ALT_i, MSR_{i,t}, DProdBR_{i,t})) \tag{11.6}$$

$$Labor_{i,t} = \int_0^t (HireR_{i,s} - RetireR_{i,s} - layoffR_{i,s})ds \tag{11.7}$$

$$Vac_{i,t} = \int_0^t (VacBR_{i,s} - HireR_{i,s})ds \tag{11.8}$$

$$HireR_{i,t} = Vac_{i,t}/E(VacFT_i) \tag{11.9}$$

$$RetireR_{i,t} = Labor_{i,t}/E(EmployT_i) \tag{11.10}$$

$$LayoffR_{i,t} = \min(\max(0, -ALabor_{i,t}),\ Labor_{i,t}/E(LayoffT_i)) \tag{11.11}$$

$$VacBR_{i,t} = \max(0, ALabor_{i,t} + AVac_{i,t}) \tag{11.12}$$

$$ALabor_{i,t} = \lambda^{(L)}(DLabor_{i,t} - Labor_{i,t-1})/LaborFT_i + (1 - \lambda^{(L)})ALabor_{i,t-1} \tag{11.13}$$

$$AVac_{i,t} = \lambda^{(V)}(DVac_{i,t} - Vac_{i,t-1})/VacFT_i + (1 - \lambda^{(V)})AVac_{i,t-1} \tag{11.14}$$

$$DLabor_{i,t} = DProdBR_{i,t}/(ALP_i \times ALT_i) \tag{11.15}$$

$$DVac_{i,t} = \max(0, VacFT_{i,t} \times ALabor_{i,t}) \tag{11.16}$$

$$Inv_{i,t} = \int_0^t (ProdCR_{i,s} - ShipR_{i,s})ds \tag{11.17}$$

$$OrderR_{i,t} = TOR \times MS_{i,t} + \sigma^{(O)} \tag{11.18}$$

$$ShipR_{i,t} = OrderR_{i,t} \times f(Inv_{i,t}/DInv_{i,t}) \tag{11.19}$$

$$DInv_{i,t} = (OPT_i + SSCov_i) \times (OrderR_{i,t}) + \sigma^{(I)} \tag{11.20}$$

$$InvCov_i = Inv_i/ShipR_i \tag{11.21}$$

$$Price_i = E(MP) \times (F_i^{(C)}) \times (F_i^{(I)}) \tag{11.22}$$

$$F_i^{(C)} = 1 + PSens_i^{(C)} \times (E(c_i^{(P)}) \Big/ E(MP) - 1) \tag{11.23}$$

$$F_i^{(I)} = (InvCov_i/MaxInvCov_i)^{(PSens_i^{(I)})} \tag{11.24}$$

$$PriceCR = ((Price_1 + Price_2)/2 - MP)/MPFT \tag{11.25}$$

The product price is determined by Eq. (11.22) according to Sterman (2000), in which three major parts take effects:

- Effect of production costs on price;
- Effect of inventory coverage on price;
- Impact of retailer/market expected price.

Figure 11.5a depicts the price determination module in the SD model. Inside the price determination mechanism, the effects of duopoly company competition (an enhancement to the original model) is incorporated into the calculation of the retailer expected price. The effect of production costs on price captures the retailer's beliefs on the production costs relative to the expected product price [see Eq. (11.23)]. Either the production cost information ($PSens^{(C)} = 0$) or the retailer's belief ($PSens^{(C)} = 1$) can be ignored depending on the values of sensitivity of price to costs. The effect of inventory coverage on price measures how the relative inventory coverage of supply chain members affects the product price. The sensitivity of price to inventory coverage serves as the exponent of the relative inventory coverage [see Eq. (11.24)], and its value is negative to reflect the relationship between inventory coverage and price (lower inventory coverage results in higher price). These two equations [Eqs. (11.23) and (11.24)] confirm to the original model in Sterman (2000). The third part of the price determination is related with the retailer/market expected price. For a particular type of product, retailers and the consumer market always maintain the belief about the expected price, mainly relying on the past price of similar product. For the simplicity concern, the price biddings among retailer, wholesaler and manufacturer are not explicitly modeled; however, to reflect the price adjustment process over time, the changing rate of market expected price is calculated by the difference of product average price and market/retailer expected price divided by a pre-defined fixed time length [see Eq. (11.25)]. In the price determination process, the experimental control variables considered are price sensitivity to production cost ($PSens^{(C)} \in [0, 1]$), price sensitivity to inventory coverage ($PSens^{(I)} \in [-1, 0]$), and manufacturer expected price (*Mfg_Price*).

11.3.2 Agent-based Simulation for Modeling Consumer Purchasing Behavior Under Duopoly Market

Figure 11.5b is the module snapshot of the consumer purchasing behavior in ABS model. Equations from (11.26) to (11.36) represent underlying mathematical relationships of the module, and Table 11.2 provides nomenclatures for variables and parameters used in those equations. For the marketing expense, it is assumed to have two aspects: advertisement and promotion. Equations (11.26) and (11.27) are used to calculate the spending for advertisement and promotion over a considered time period (i.e. a period for a certain marketing strategy). The amount of marketing budget is decided according to the company's revenue. The Chief Marketing Officer (CMO) Council report (2010) demonstrates a direct relation between marketing budget and revenue for various companies. Based on our case study (i.e. soft drink duopoly), the corresponding percentage of marketing investment is selected. An adjustment factor is introduced in these two equations to ensure that a realistic scenario (e.g. order of magnitude) can be achieved. The market spending rate is then derived [see Eq. (11.28)] by incorporating the adjustment time for spending market budget into the calculation. The decision

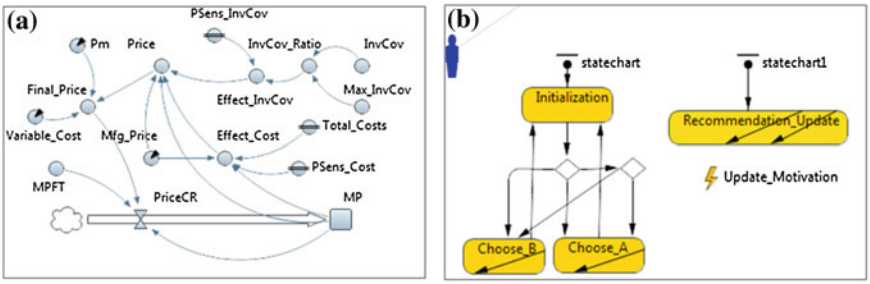


Fig. 11.5 **a** Price determination module in SD model (*left*), **b** consumer purchasing behavior in ABS model (*right*)

Table 11.2 Nomenclature for agent-based model

Notation	Explanation	Notation	Explanation
MB	Marketing budget	ρ	Co-state parameter
Ad	Advertisement intensity	δ	Co-state factor
Pm	Promotion depth	$C^{(s)}$	Marketing sunk cost of the duopoly companies
K	Adjustment factor	*Inf*	Follower tendency influence
AdS	Spending rate on advertisement	*M*	Customer purchasing motivation function
PmS	Promotion spending rate	*SensP*	Price sensitivity
AdjTimeMS	Adjustment time for marking budget spending	*SensPm*	Sensitivity of consumer to promotion
MSR	Marketing spending rate	*SusAd*	Susceptibility of consumer to the advertisement
$\omega_i, i = 1, 2, 3$	Weights of market force effects	*Ft*	Follower tendency
Inter	Interaction effect between two duopoly companies	s, m	Price sensitivity parameters
MF	Marketing force	I_a, I_p, I_f	Initial value of *SusAd*, *SensPm*, and *Ft*.
F	Total marketing force	*Mfg_Price*	Manufacturer expected price

Subscripts i and t are omitted. i is player index (i = 1, 2), t represents simulation time

variables considered in the marketing strategy are marketing budget (*MB*), advertisement intensity (*Ad*), and promotion depth (*Pm*).

Marketing force concept in this work has been adopted from extended Lanchester model (Naik et al. 2005) and is depicted in Eq. (11.29). The marketing force depends not only on the weight of advertisement intensity, promotion efforts (e.g. frequency and depth), but also on their marketing strategy interactions that are discussed in details in the next paragraph. The marketing force is the leading power, which influences the consumer's perception (e.g. sensitivity of promotion, susceptibility of advertisement) of a particular product. In this chapter, these relationships are captured in Eqs. (11.30)–(11.32). As the market force is dynamically updated through the simulation run, the consumer's perception is also updated according to the change of market force. This assumption implies that if a company loses most of the market, it would have to sacrifice even more to win back the market share.

Another important feature in the ABS model is that we explicitly incorporate the marketing interaction effects between companies. These marketing interactions include the binding constraints on the sum of expenditures on the advertisements and promotions, as well as the segregation of locations and communication channels expressed in terms of expenses (Naik et al. 2005). In this work, the mathematical formulation is based on these concepts, where the interactions for each pair of activities are explicitly modeled. To take the strategic foresight of manager into account, co-state dynamics in Eq. (11.33) is adopted, and the interaction effects between companies are formulated as the co-state variables. The values of co-state variables in the next time point are captured by the differential equation given the current interaction effects. Then, the sunk cost is calculated [see Eq. (11.34)], which incurs due to the strategic interactions between duopoly companies. The case study presented in Sect. 11.5.1 provides more details on these interactions in the context of a soft drink duopoly competition.

In a consumer market, consumers make the adoption decision based on various factors from both the companies and environment, such as unit price, advertisement, promotion, quality, and word-of-mouth recommendations. In our simulation model, it is assumed that an agent (i.e. consumer) becomes an adopter of a particular product based on the motivation function incorporating effects of four factors—price sensitivity, advertisement influence, promotion sensitivity, and acquaintance influence. Based on the model in Zhang and Zhang (2007), we proposed an improved formula to calculate the consumer motivation to purchase brand i at time point t, in which the motivation value is decided by the following three attributes of price, advertisement intensity, and agent influence exerted by other agents (consumers). The enhancements made in this work are as follows:

- Incorporation of a social network structure to represent interactions among agents;
- Incorporation of advertisement and promotion factors to mimic more realistic decision making process.

In our study, a scale-free social network model called Barabasi-Albert Model (BA model, also known as Preferential Attachment Model) (Albert and Barabasi 2002) is built to represent the social relationships of customers for the artificial market. The BA model reflects the "rich-get-richer" phenomenon in societies and the degree of nodes follows a power-law distribution, in which the probability of a new node connecting to an existing node is proportional to the degree of it. To incorporate the advertisement and promotion effects from marketing activities into the consumer purchasing decision, the price sensitivity, susceptibility to advertisement, promotion-sensitivity and follower tendency, have been set to associate with price, advertisement, promotion, and recommendation influence, respectively. The initial value of susceptibility to advertisement, promotion-sensitivity and follower tendency are pre-set at the beginning of simulation run. The price sensitivity is an exponential function of the difference between the real price of a product and the expected average price of the product [see Eq. (11.35)]. In this equation, s is a price parameter ($s > 1$), and takes the same values for the similar competitive types of product, m is a constant and its value is based on an agent's socio-economic attributes (e.g. millionaires are less price sensitive than unemployed persons). The consumer purchasing motivation function is calculated in Eq. (11.36), which decides the product selection of consumers. It is assumed that agents will always select a product having a higher motivation value, and randomly choose one if the motivation values are equal.

$$AdS_i = K \times MB_i \times Ad_i \tag{11.26}$$

$$PmS_i = K \times MB_i \times Pm_i \tag{11.27}$$

$$MSR_i = (AdS_i + PmS_i)/AdjTimeMS \tag{11.28}$$

$$MF_i = \omega_1 Ad_i + \omega_2 Pm_i + \omega_3 Ad_i Pm_i + Inter_i \tag{11.29}$$

$$SusAd_i = MF_i \times I_a \tag{11.30}$$

$$SensPm_i = MF_i \times I_p \tag{11.31}$$

$$Ft_i = MF_i \times I_f \tag{11.32}$$

$$\begin{pmatrix} \overline{inter_1} \\ \overline{inter_2} \end{pmatrix} = \begin{pmatrix} \delta_1 & \delta_1\delta_2 \\ \delta_2\delta_1 & \delta_2 \end{pmatrix} \begin{pmatrix} \rho + F & 0 \\ 0 & \rho + F \end{pmatrix} \begin{pmatrix} inter_1 \\ inter_2 \end{pmatrix} - \begin{pmatrix} Price_1(1 - Pm_1) \\ Price_2(1 - Pm_2) \end{pmatrix} \tag{11.33}$$

$$C^{(s)} = \sum_{i=1}^{2} MB_i \times Inter_i \tag{11.34}$$

$$SensP_i = -s^{Price_i \times (1 - Pm_i) - (Price_1 + Price_2)} + m \tag{11.35}$$

$$M_i = SensP_i \times Price_i(1 - Pm_i) + SusAd_i \times Ad_i + SensPm_i \times Pm_i + Ft_i \times Inf_i \tag{11.36}$$

11.3.3 Payoff in Simulation-based Duopoly Game

The total net profit serves as the payoff of simulation-based game, which is calculated in Eq. (11.37). The cost items constituting the payoff function based on the simulation outputs are depicted in Table 11.3. All different cost items across the production, logistics and marketing activities are considered in the payoff function, and the time length to calculate all the cost items is the total simulation replication length. After the simulation run, the outputs are collected to calculate the net profit earned for each company. A payoff matrix is then constructed based on the outputs and is used to approximate the best response (i.e. equilibrium) of the duopoly game, which will be discussed in Sect. 11.4.

$$Payoff_i = TRev_i - (C_i^{(P)} + C_i^{(R)} + C_i^{(I)} + C_i^{(B)} + C_i^{(T)} + C_i^{(M)} + C^{(S)}) \tag{11.37}$$

Table 11.3 Nomenclature for game payoff components

Payoff components	Descriptions
$TRev_i$	Total revenue for product i
$C_i^{(P)}$	Total production cost for product i
$C_i^{(R)}$	Total raw material purchasing cost for product i
$C_i^{(I)}$	Total inventory cost for product i
$C_i^{(B)}$	Total backlog cost for product i
$C_i^{(T)}$	Total transportation cost for product i
$C_i^{(M)}$	Total marketing spending for product i

11.4 Simulation-based Game Solving and Analysis

In this section, a detailed simulation-based GSA procedure proposed in this chapter will be discussed. The intent of the proposed procedure is to make the problem tractable by restricting the profile strategies that each company is allowed to play without losing the generalization from the original game. Large strategy spaces consist of continuous and multi-dimensional action sets, while the perfect information assumption is assumed to hold to reduce the problem complexity for analysis. Due to the symmetric property of the game, two agents are assumed to have identical behavior possibilities, and be exposed to the same customer market. Before discussing the details of the GSA components and procedure, notations regarding a normal form game, simulation-based game and the equilibrium concepts are introduced first.

11.4.1 Setup and Motivation of Simulation-based Game Solving and Analysis

A *normal form game* can be formally expressed as $\Gamma = [I, \{s_i, \Delta(s_i)\}, \{u_i(s)\}]$, where I refers to the set of players and $I = \{1, 2\}$ in our study; s_i and $\Delta(s_i)$ denotes the pure and mixed strategy for player i ($i \in I$) respectively; $u_i(s)$ is the payoff function of player i when strategy profile s has been selected. An important variable frequently used in analyzing normal form game is *regret* of a profile $s \in S$, denoted by $r(s)$, which is calculated in Eq. (11.38).

$$r(s) = \max_i \max_{s_i'} u_i(s_i', s_{-i}) - u_i(s) \tag{11.38}$$

In Eq. (11.38), $s_i' \in \{S_i - \{s_i\}\}$ and $s_{\text{-}i}$ represents for a strategy profile other than that of player i. Next, definition regarding game solution is given as follows: a *Nash equilibrium* of the normal-form game is a strategy profile $s \in S$ such that for every player $i \in I$, Eq. (11.39) holds.

$$u_i(s) \geq u_i(s_i', s_{-i}), \forall s_i' \in S_i \tag{11.39}$$

In this chapter, Nash equilibrium, equilibrium, and game solution terms are used interchangeably. Furthermore, the *symmetric game* setting is also considered, in which the following two conditions need to be satisfied:

- $S_i = S_j$ for all players $i, j \in I$;
- $u_i(s_i, s_{-i}) = u_j(s_j, s_{-j})$ for every $s_i = s_j$ and $s_{-i} = s_{-j}$.

In addition, the terms of simulation-based game and empirical game are used interchangeably because they essentially convey identical meanings. A *simulation-based game* is defined that the player's payoff is specified via simulation models, and the definition of *empirical game* is focused on estimating the payoff matrix using simulation outputs (Vorobeychik 2008). In the empirical game setting with a large number of strategy profile and noisy samples involved, calculating the exact Nash equilibrium is sometimes intractable. Another way of approximating it is applying ε-Nash equilibrium (ε: tolerance), which is a profile $s \in S$ satisfying Eq. (11.40) for every player $i \in I$.

$$u_i(s) + \varepsilon \geq u_i(s_i',\ s_{-i}), \forall s_i' \in S_i \tag{11.40}$$

As the game is constructed in simulation, we differentiate two types of payoff: the *true payoff* existing in a real practice duopoly and the *estimated payoff* obtained from simulation outputs. When constructing an empirical payoff matrix, a simulation model will be run to obtain noisy samples for each pure or mixed strategy profile. The noisiness in the sample includes the randomness from the simulation experiments as well as the players' mixed strategies (Vorobeychik 2010). Empirical game is the one, which maintains the same strategy profiles for all players while the payoffs of them involve noise. For each specific profile of any player in an empirical game, the payoff is an estimate value by taking arithmetic mean of multiple data points from the noisy sample as shown in Eq. (11.41).

$$\hat{u}_{i,n}(s) = \sum_{j=1}^{n} U_{ij}(s) \bigg/ n \tag{11.41}$$

The equation shows an estimate of payoff to player i for profile strategy s based on n samples. From now on for the terminologies used in our discussions, readers are suggested to refer to Table 11.4.

The academic challenge of solving such a game is that the constrained simulation and experimental resource cannot afford the enormous number of strategies and samples. According to the discussion in Sect. 11.3, the duopoly game includes totally 12 strategic factors for each player: if every single strategic factor takes only two levels, the total number of profiles in the entire profile set under symmetric game setting is $(2^{12})^2/2 = 8{,}388{,}608$. Assuming each simulation replication takes 1 s and only 10 replications are taken for each individual profile, the total time needed to complete the simulation of the entire profile set would be 2.66 years, which is unrealistic to perform in practice.

Table 11.4 Clarification of terminologies used

Name	Definition	Explanation
Aggregated strategic factor	The factor including aggregated information of other factors	Production factor, Logistics factor,
Detailed strategic factor	The factor decomposed from aggregated strategic factor	Order lead time, safety stock coverage decomposed from logistics factor
Strategic factor levels	The different levels (i.e. values) that a factor can achieve/attain	(*H*) for high level of production factor
Strategy	Combination of different levels for a group of strategic factors*	(*H*, *L*, *L*, *H*)*
Profile	Combination of strategies chosen by game players	$\{(L, L, H, H)_1, (H, L, H, L)_2\}$ is one profile for a two-player game
Solution profile	Players' profile obtained when game reaches the equilibrium	Element(s) in the profile set
Solution payoff	Players' payoff obtained when game reaches the equilibrium	Element(s) in the payoff matrix
True payoff	The ideal payoff for the game player	$u(s)$ with respect to profile s
Estimated payoff	The estimated payoff value obtained from simulation	$\hat{u}_{i,n}(s)$ with respect to profile s by running n simulation samples

**H* high, *L* low; *Strategic factor* decision variable

The above computational challenge motivates development of the GSA procedure in this chapter. As it is impractical to construct a comprehensive payoff matrix and achieve the exact game equilibrium(s) by involving all strategic factors (and their levels), targeting on the critical factors that can approximate the true equilibrium becomes the major undertaking. As the number of profiles is reduced, the sample size for each profile can be increased accordingly. The trade-offs between *strategy refinement* and *data sampling* is: given a fixed amount of simulation/ experimental resources, exploring more profiles decreases the number of samples that can be chosen, which may influence the accuracy of estimated game payoff by the end; while more samples will restrict the span of profiles to be selected, which may rule out the key strategies that will impact the game solution eventually. Other than the strategy refinement and data sampling, a game solving engine/algorithm and performance evaluation criteria are also needed to complete the GSA.

11.4.2 Components and Procedure of Simulation-based Game Solving and Analysis

To resolve the formulated simulation-based game, four components are required:

- First of all, an approach to explore and refine the strategy space is needed. As discussed before, some strategies are more significant to determine the game equilibrium than others. Our objective here is to explore those critical strategies in a more detailed manner so that insights can be gained on how the key strategic factors can impact on the game equilibrium.

- Second, a method to decide the sampling procedure is needed. As known, sampling cost and information gain are always the trade-offs during the sampling procedure. Given the sampling resource availability and capability, the sampling procedure should be able to achieve the maximum information gain so as to better approximate the true game payoff.
- Third, a game solving engine is needed, which will allow us to find equilibrium(s) for the simulation-based game under different initial game settings (e.g. initial strategy profile, problem scenario). The game solution should include pure, mixed or both types of equilibriums.
- Forth, evaluation criteria for assessing the performance of GSA procedure is needed, which will capture the main features of the GSA procedure by dealing with the game equilibrium results. The evaluation criteria should also contain the assessments of major equilibrium properties (e.g. weakness, strictness, stability, and robustness).

We first formulate an algorithm, which depicts how these four components mentioned above work together to solve and analyze the simulation-based game. Then detailed contents on each component are discussed. Note that each round of the GSA procedure run is called an *iteration*. The GSA procedure includes the following major steps:

Step 1 Develop an initial game *profile set* by selecting strategic factor levels, then choose an initial *sample size* for each profile and set g equals to 1.

Step 2 Run the simulation model based on the selected profile set and sample size, construct the *empirical payoff matrix* according to the simulation outputs.

Step 3 **(Game Solving)** Solve the game for pure strategy equilibrium by improving the *unilateral deviation set* for each player one after the other until no more improvements can be obtained.

Step 4 **(Strategy Refinement)** Employ *design of experiments* (DOE) technique to decide the statistical significance of each *aggregated strategic factor* with respect to the game payoff. Then, if g equals to 1, go to Step 4.1; if g equals to 2, go to Step 4.2.

Step 4.1 Include all the *detailed strategic factors*, which are decomposed from the current *significant* aggregated strategic factor, into the *refined profile set*; eliminate the insignificant aggregated strategic factor(s) from the *refined profile set*. If no more detailed strategic factors can be included, go to Step 5 and set g equals to 2.

Step 4.2 Include more strategic factor levels into the *refined profile set* for the next *iteration*, go to Step 5. If no more levels for each strategic factor need to be added, terminate the GSA process and go to Step 6.

Step 5 **(Data Sampling)** Given the significant strategic factors and their levels in the *refined profile set*, decide the sample size for each profile using the *enhanced ECVI sampling* approach. Go to Step 2.

Step 6 **(Performance Evaluation)** Based on the game equilibrium results, calculate values for all the *evaluation criteria* inside and between GSA iterations, and summarize the results.

Steps 1 and 2 are mainly for algorithm initialization and payoff generation, respectively. An indicator variable g is used in Step 4, which represents the refinements of either strategic factors ($g = 1$) or factor levels ($g = 2$). Provided that a reasonable experimental time and cost can be spent on the simulation experiments, the trade-offs between the strategy refinement extent and data sampling size always exist. Table 11.5 provides comparison results with varying numbers of strategy refinement and sampling size given a fixed affordable experimental time (i.e. 5 days) for the simulation run. The lower limit of the experimental cost is bounded by ensuring a minimum degree of strategy refinement and sampling size, while the upper limit is related with the total affordable experimental cost. As shown in Table 11.5, if each experimental iteration is selected to be 5 days, a total of four strategic factors can be selected to ensure a reasonable number of samples (i.e. 150) in the experiments.

The strategy refinement method essentially seeks to find out in which order and with what specific strategic factor levels to include the strategies to the simulation-based game analysis. It is slightly different to the strategy exploration problem in Jordan et al. (2008), with the modification of the word "refinement" that is tightly related with both the game strategy and simulation modeling details. As noted before, each strategic factor (e.g. production) involves different detailed aspects (e.g. labor control, raw material procurement). The strategic factors that are more significant than others should be considered with priority in the simulation testbed and also decomposed into more detailed levels for analysis. The purpose of doing

Table 11.5 Trade-offs between strategy refinement and data sampling

Total strategic factors	No. of strategies for each player (level = 2)	No. of profiles to be evaluated	Affordable experimental time limit (days)	Time per simulation replication (seconds)	No. of samples affordable for each profile
1	2	3	5	20	7200
2	4	10	5	20	2160
3	8	36	5	20	600
4	16	136	5	20	158.82
5	32	528	5	20	40.90
6	64	2080	5	20	10.38
7	128	8256	5	20	2.62
8	256	32896	5	20	0.66
9	512	131328	5	20	0.16
10	1024	524800	5	20	0.04

so is to approximate the game equilibrium without evaluating all the strategy profiles, which is time-consuming, cost-inefficient, and even intractable. The strategy refinement process, which starts from an aggregated level and then moves to a more detailed level, is set as follows:

- For the initial experiment, the focus of the profile set (simulation inputs) is only at the aggregated strategic factors (e.g. production, logistics), and multiple (e.g. 2) levels of these factors are selected for experimental study.
- *Design of experiments* technique is then used to identify the critical strategy profiles by analyzing the simulation outputs. Figure 11.6 depicts the process, in which the inputs to the experimental design is the different levels of strategic factors and the empirical payoff matrix generated from simulation outputs, while the outputs of the experimental design are the factors that have significant impacts on the game payoff.
- Then, for those critical strategies, more insights on how different values of strategic factors impact the game payoff are investigated via partitioning the factors into detailed factors or levels depending on the requirements. Then, we treat each strategic factor or level as the input to the simulation for the next experiment iteration.

The above mentioned process (i.e. empirical payoff generation via game simulator, identification of significant factors via experimental design) is applied iteratively in GSA procedure. During the iterative process, game is solved and the immediate results are used to find the corresponding profiles for sampling. It is noted that under different simulation scenarios, the outputs of experimental design may be different. In addition, various experimental design techniques may be applied as long as they provide better insights into the analysis. This work employs a standard two-level full factorial experimental design technique as a pilot study for strategy refinement.

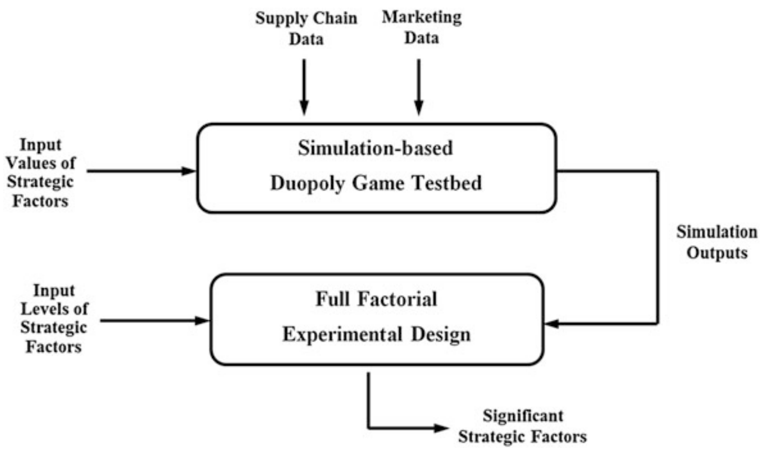


Fig. 11.6 Experimental design for strategic factor refinement via simulation

For sampling significant profiles, an approach named *estimated conformational value of information* (ECVI) in Walsh et al. (2003) has been enhanced in this chapter. The ECVI measures the degree to which further samples would reduce the estimated error (denoted by $\hat{e}(x)$) of the current equilibrium solution [see Eqs. (11.42) and (11.43)].

$$\hat{e}_{i,p}(x(s)) = \hat{u}_{i,p}(s) - u_i(s) \tag{11.42}$$

$$ECVI(q|i,s,p) = E_{q|p}[\hat{e}_{i,p}(x(s)) - \hat{e}_{i,p.q}(x(s))] \tag{11.43}$$

In Eqs. (11.42) and (11.43), s represents a strategy chosen to conduct sampling, p and q refer to the number of data points being sampled and to be sampled, respectively. The maximum information gain is achieved by selecting the maximum value of ECVI, which also indicates the best choice of samples. This method has been chosen in our study as it has been approved to show significant improvement over the uniform sampling method. While the criteria for stopping sampling and the tradeoff between the sampling cost and information value gain are not discussed in details in Walsh et al. (2003), our work addresses them explicitly. The sampling cost mainly depends on the simulation replication length, and the information value gain refers to how important more samples can help to make an accurate decision. In this chapter, two separate items in ECVA are classified in the GSA procedure:

- A pre-selected threshold value of affordable sampling size, which is the maximum number of samples that can afford to run for each profile based on the experimental resource availability.
- A lower limit of *ECVI gain*, which is designed by user and aimed to ensure the game solution quality.

Under the two criteria discussed above, we want to find the corresponding sample size either satisfying the lower limit of ECVI gain (denoted by $ECVI_s^{(L)}$), or reaching the limit of sampling capability (denoted by N_s), as shown in Eq. (11.44).

$$p + q = \min(N_s,\ ECVI_s^{(L)}) \tag{11.44}$$

This enhancement provides flexibility to users, where they can select their own threshold values depending on the experiment requirements. In our experimental study, we have applied this approach to eliminate the twisted sample values (the extreme low and high values), which constitute about 10 % of all data samples.

Integration of strategy refinement and data sampling discussed so far in this section contributes to the uniqueness of the proposed GSA approach. This integration allows us to combine the advantages of both, as well as to avoid the potential drawbacks of spending additional simulation resources for sampling all profiles. The next step in our procedure is to input the selected game strategy profile and sample size into the simulation-based game testbed. The simulation outputs are then collected to construct the empirical payoff matrix. Then, we apply a game solving engine to calculate the pure Nash equilibrium for the duopoly players. The game solving engine computes the equilibrium by improving the *unilateral deviation* set in Eq. (11.45) for each player one after another, until no more payoff gain can be obtained.

$$D_i(s) = \{(\breve{s}_i, s_{-i}) : \breve{s}_i \in S_i\},\ i = 1, 2 \tag{11.45}$$

This is a traditional approach, but still the most effective and efficient way to obtain the pure Nash equilibrium. As the empirical payoff matrix always involves variations, an ε-Nash equilibrium concept [see Eq. (11.40)] is used to ensure that the potential optimum solutions are included during each experiment iteration.

As the game solution involves variations due to different reasons such as limited simulation/experimental resources and sampling errors inherent to simulation, proper criteria on assessing the GSA procedure has been developed in this chapter. As mentioned earlier, the GSA procedure stops when no more iteration (e.g. strategy refinements) can be established. As each experiment iteration proceeds and the simulation gains more fidelity (details), we intend to find (1) whether the equilibrium stays unchanged or evolves to be better (e.g. strictness vs. weakness), (2) how the modeling details can impact the game payoff, and (3) how sensitive the equilibrium(s) are to the disturbances. The evaluation criteria developed in this work focus on the following aspects:

- *Confidence intervals* of the game equilibrium(s) for examining the closeness of estimated and true payoffs [see Eq. (11.46)];

$$\Pr(\hat{u}_{i,n}(s) - \theta < u_i(s) < \hat{u}_{i,n}(s) + \theta) = 1 - \alpha \tag{11.46}$$

- *Statistical test* (i.e. two-sample t test) for evaluating the differences between solution profile and its neighboring profiles;
- *Statistical test* (i.e. two-sample t-test) for evaluating the differences between solution payoffs over iterations;
- Experimental studies on the stability of the game equilibrium(s): the equilibrium stability concepts applied here are originated from Szidarovszky and Bahill (1998), and we define three types of stability as follows:
  - *Asymptotic stability* with respect to game equilibrium refers to that for a given initial game state (i.e. players' initial profile), the player payoff for the solution profile eventually converges to the solution payoff.
  - *Marginal stability* with respect to game equilibrium is the one that for a given initial game state, the player payoff for the solution profile converges to a region containing the considered solution payoff and its tolerance.
  - *Instability* with respect to game equilibrium refers to the players' profile that does not belong to the above two categories.

11.5 Experiments and Results

11.5.1 Experimental Setup in Soft-drink Duopoly Scenario

Under the current market scanner, the soft drink industry exhibits a classic example of duopolies involving integrated supply chain and marketing activities. Cola wars between The Coca-Cola Company® and PepsiCo Inc® and related literature works (Mckelvey 2006) have served as a basis for our case study. The two companies together account for about three-quarters of the total soft drink market share. In fact, the industry has high operational overlap since different suppliers and manufacturers (e.g. producers and bottlers) possess similar impetus of sales and profits along the supply chain, and in the market side a similar customer base is shared for the duopoly companies. While the soft drink industry as a whole enjoys positive economic profits among all of its members, the ultimate goal for the industry should be to create a win–win situation for both the manufacturers as well as the customers.

As mentioned earlier, both Coca-Cola Company® and PepsiCo Inc® mainly trade on supply chain and marketing values, and invest substantial portion of their revenues in those areas. Modeling of the major activities in those areas has been discussed in Sect. 11.3. Different values of the decision parameters for the proposed simulation model depict the various scenarios encountered in the soft drinks duopoly. Table 11.6 shows the strategic factor values used in the experiments of this chapter, and the length of simulation replication run is about 3 months (100 days). We then estimate a payoff matrix through the constructed normal-form simulation-based game, with the emphasis on the strategies mentioned in Table 11.6.

Table 11.7 shows the strategic factors and levels involved in each experiment iteration.

To balance the trade-offs between strategy refinement and data sampling, 16 strategies for each player (four strategic factors) and 70 initial data samples for each profile are selected during each experiment iteration (the total number of profile is 16*16 = 256). As the considered game is symmetric, only the upper triangular of the strategy matrix is used for sampling, which is equivalent to 136 ([(16*16)-16]/2 + 16 = 136) profile sets. After applying the modified ECVI data sampling approach, samples with roughly 10 upper and 10 lower extreme values have been eliminated for each profile. So the effective sample size in our experiment is 50. As each iteration may involve different strategic factors (aggregated or detailed), notation $(S_m^{(k)}, S_n^{(k)})$ is used to represent the profile information for player A selecting strategy m $(m = 1, 2, \ldots, 16)$ and player B selecting strategy n $(n = 1, 2, \ldots, 16)$ during kth iteration.

Table 11.6 Strategic factors and levels used in experiments

Aggregates strategic factor	Detailed strategic factor	Strategic factor levels			
		Low*		High*	
		L	ML	MH	H
Manufacturing	Vacancy creation time (days)	1		5	
	Average time for layoff labors (days)	3		7	
	Labor fulfillment time (days)	4		12	
	WIP fulfillment time (days)	1		3	
Logistics	Inventory fulfillment time (lead time) (days)	2	6	10	14
	Raw material transportation lead time (days)	1	3	5	7
	Product safety stock coverage (days)	2	6	10	14
	Raw material inventory coverage (days)	1	3	5	7
Pricing	Price sensitivity to production cost ($\in [0, 1]$)	0.1		0.9	
	Price sensitivity to inventory coverage ($\in [-1, 0]$)	-0.1		-0.9	
	Manufacturer expected price ($)	1		2	
		5 % of revenue		15 % of revenue	
	Marketing budget (MB) ($)	5 % of revenue		15 % of revenue	
Marketing	Promotion depth (% of MB, uniform distributed)	(0.1, 0.2)	(0.2, 0.3)	(0.1, 0.2)	(0.2, 0.3)
	Advertising intensity (% of MB, uniform distributed)	(0.1, 0.2)	(0.2, 0.3)	(0.1,0.2)	(0.2, 0.3)

*L: low, ML: medium low, MH: medium high, H: high

Table 11.7 Strategic factors used over GSA iteration in experiments

Iteration	Strategic factors	Strategic factor levels*	Iteration	Strategic factors	Strategic factor levels*
1st	Manufacturing	L/H	2nd	Advertising intensity	L/H
1st	Logistics	L/H	3rd	Raw material inventory coverage Product safety stock coverage	L/ML/MH/H
1st	Pricing	L/H	3rd	Raw material transportation lead time Inventory fulfillment time	L/ML/MH/H
1st	Marketing	L/H	4th	Raw material inventory coverage Product safety stock coverage	L/ML/MH/H
2nd	Raw material inventory coverage Product safety stock coverage	L/H	4th	Promotion depth	L/ML/MH/H
2nd	Raw material transportation lead time Inventory fulfillment time	L/H	5th	Raw material inventory coverage Product safety stock coverage	L/ML/MH/H
2nd	Promotion depth	L/H	5th	Advertising intensity	L/ML/MH/H

*L: low, ML: medium low, MH: medium high, H: high

11.5.2 Experimental Results

In this section, we describe the experimental results and demonstrate the effectiveness of the proposed GSA procedure under the hybrid simulation framework. For the limited space, only pure strategy equilibrium(s) are analyzed in this section.

Figure 11.7 depicts the percentage of game equilibriums computed for all five iterations, in which the horizontal axis represents the solution percentage and the vertical axis represents the payoff tolerance. As it is a duopoly game, it is highly believed that the solution profile has the symmetric strategy for the two players (i.e. $(S_{n_1}^{(k)}, S_{n_2}^{(k)})$ with $n_1 = n_2$). That's the reason why we only draw the symmetric strategy in Fig. 11.7, and notify other potential solution strategies as "Others". In Fig. 11.7, we observed that as the payoff tolerance increases within each iteration, the empirical game tends to involve more equilibriums than the case under zero tolerance. As the tolerance value is highly related with the sample size of each

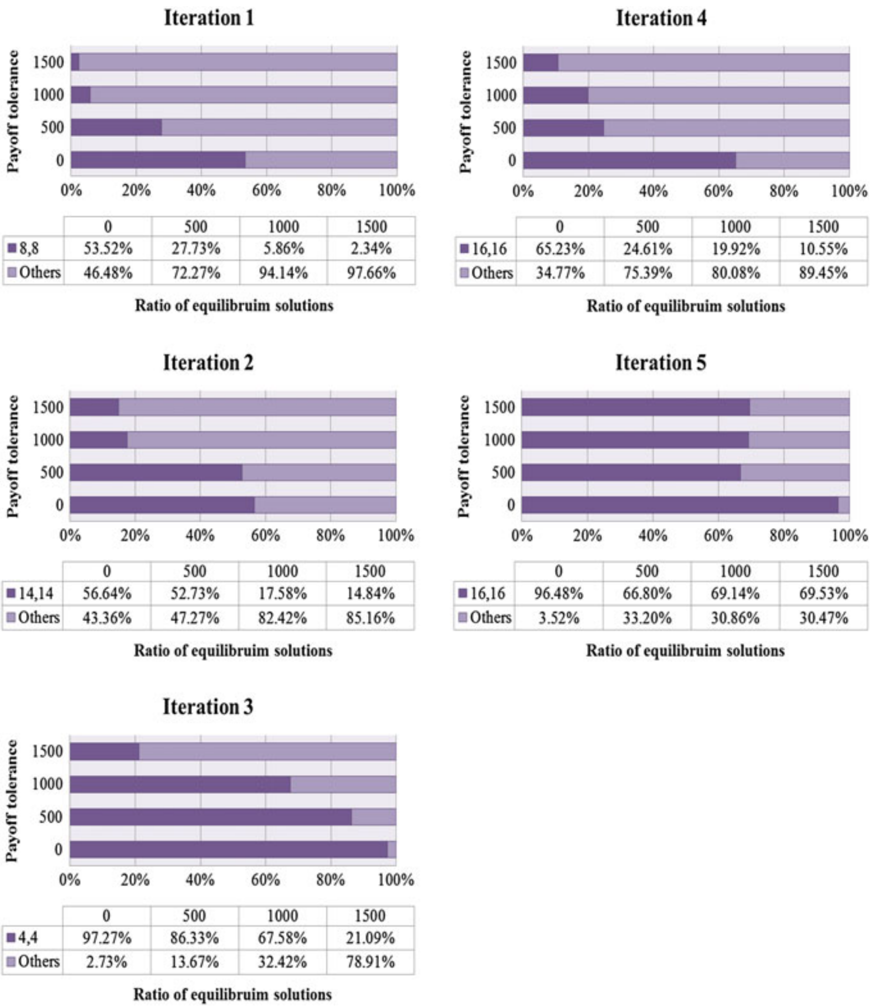


Fig. 11.7 Evolution of game equilibriums over GSA iterations

profile, it is difficult to reduce the tolerance by sampling more data (sampling cost is limited). However, the suspected solution profile with its neighborhood strategies, which only involves roughly 8–12 data points, can be extracted out and sampled with more data points. Another observation from Table 11.8, which conforms to our intuition, is when the sample size enlarges from 50 to 500, the half width of confidence interval for each potential solution payoff reduced. As the iteration proceeds (from iteration 1 through 5), the half width of confidence interval (CI) also decreases, which indicates the estimated equilibrium is closer to the true equilibrium. However, under the sample size of 500, the decreasing trend of the CI values is not as salient as that for the case with the sample size 50 or under.

Table 11.8 Comparisons of solution profiles and payoffs over GSA iterations

Iteration	ES_1*	ES_2*	Payoff for player 1				Payoff for player 2			
			Sample size = 500		Sample size = 50		Sample size = 500		Sample size = 50	
			Mean	HW	Mean	HW	Mean	HW	Mean	HW
1st	8	8	23967	449	23333	1257	23922	467	23265	1184
2nd	14	14	24438	450	23952	1129	24452	461	24401	1073
3rd	4	4	24854	422	25829	924	24845	441	24539	822
4th	16	16	25428	408	23778	915	25505	415	25671	841
5th	16	16	26178	397	26964	729	26305	408	28235	866

*ES_i: Equilibrium for player i

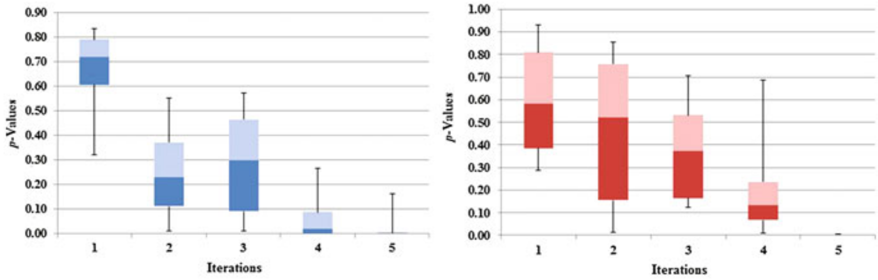


Fig. 11.8 Box-plots for the *p*-values of two sample *t*-test on solution profile with its neighbor profiles over GSA iterations: player A (*left*); player B (*right*)

To evaluate the game equilibrium robustness (weakness vs. strictness), a group of 10 samples close to the estimated Nash equilibrium (its neighborhood that have the similar payoff values with it) have been selected in each iteration. Two sample *t*-tests (hypothesis testing $H_0 : \mu_1 = \mu_2$, $H_1 : \mu_1 < \mu_2$) are then performed on each pair of selected data samples, followed by the two-tailed *p*-value calculation. Figure 11.8 organizes the calculated *p*-values into the box-plot, in which a reduced trend of major portion (25–75 %) and the median of data are observed over iterations for both player A (left) and B (right). In other words, initially (iteration 1 or 2) the game equilibrium is not significantly different from its neighborhood values; while after several iterations, the game equilibriums are almost all significantly different from their neighborhood values (iteration 5). From the results of box-plots, a conclusion can be made: the game equilibrium(s) evolves from weak to strict during iterations of the GSA procedure.

Another statistical test involves the equilibrium comparisons over different iterations. As there seems an increasing trend of equilibrium payoff over iterations, this test helps to identify the significant differences between each pair of equilibrium payoffs. The one-sided hypothesis testing is constructed with $H_0 : \mu_1 = \mu_2$, $H_1 : \mu_1 < \mu_2$; and the comparisons are performed between iterations. Figure 11.9 shows the comparison results in a bar chart, where the horizontal axis numbers

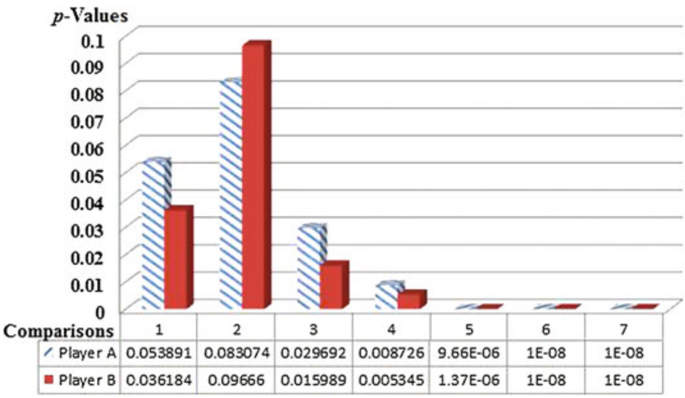


Fig. 11.9 *p*-values for comparisons of solution payoffs between iterations

Table 11.9 Comparisons of profile stability under tolerance $\varepsilon = 1500$

Iteration	Ratio of AS* profiles (%)	Ratio of MS* profiles (%)	Ratio of instable profiles (%)
1st	2.34	71.48	26.17
2nd	14.84	57.81	27.34
3rd	21.09	58.20	20.70
4th	10.55	76.56	12.89
5th	69.53	15.63	14.84

*AS: Asymptotic Stable, MS: Marginal Stable

(1 through 7) correspond to comparison groups of (1 vs. 2), (2 vs. 3), (3 vs. 4), (4 vs. 5), (3 vs. 5), (2 vs. 5), and (1 vs. 5), respectively. The one-tailed *p*-values of all comparisons are listed at the bottom of Fig. 11.9. From the figure, it is observed that every iteration improves the game equilibrium payoff with different extents, while the equilibrium result of the last iteration (5) is significantly larger than those of all the previous iterations.

Lastly, experiment results on game stability issues are provided in Table 11.9. To ensure the steady state, each player was deciding its strategies repeatedly for an extremely large amount of times (e.g. 2000 times/steps in our study). From Table 11.9, a decreasing trend of instable area is observed through iterations 1 to 5 (from 26.17 to 14.84 %). Considering the stability set from iterations 1 to 5, the asymptotic stability area increases from 2.34 to 69.53 %, and the marginal stability area decreases from 71.48 to 15.63 % (iteration 4 is an abnormal case and needs further investigation). A larger stable area brings a greater portion of points that can eventually converge to the game equilibrium or its acceptable tolerance region. So, the players or game analyst will have an increased confidence to believe that the calculated equilibrium could be achieved.

In addition to the game-theoretic analysis, the proposed simulation framework can be used to help the company managers gain useful insights through comparative analysis. For example, Fig. 11.10 summarizes the simulation state comparisons

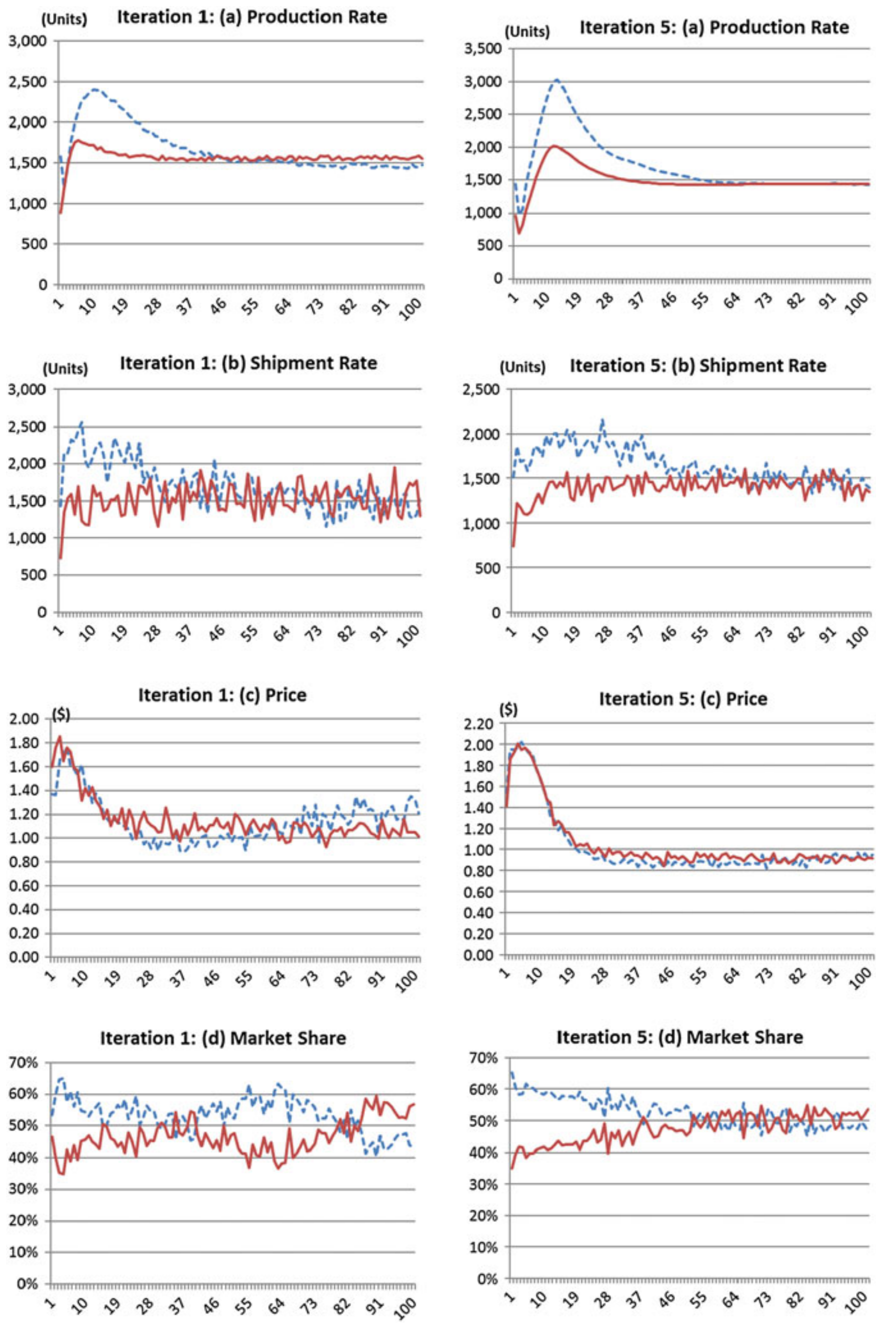


Fig. 11.10 Simulation steady state comparisons of game equilibrium on player A (*dotted*) and player B (*solid*) for different aspects between iterations 1 (*left*) and 5 (*right*)

between equilibriums of iterations 1 and 5. Note that the horizontal axis in all figures is the simulation run length. It is observed that the warm-up period takes roughly 40–50 days, so the simulation replication length has been set as 100 days (horizontal axis) to reach the system steady state. In addition, the random noises and disturbances are intentionally created to test how both players perform. As observed in Fig. 11.10, although the averages of all the outputs are mostly identical, the simulation steady state of game equilibrium in iteration 5 (right figures in Fig. 11.10) in general is more stable and involves less variations than the one in iteration 1 (left figures in Fig. 11.10) given the same amount of noises and disturbances. The weak dominance in iteration 1 is more sensitive and may change between the two players over time depending on the disturbances. The changing trend tends to last long, and the changing amount tends to accumulate high before company takes appropriate actions to compensate. Under the strict equilibrium, the dominance is shared by the two players and is not quite sensitive to the disturbances.

11.5.3 Summary of Experiments

In summary, given the duopoly case study scenario discussed in Sect. 11.5.1, as the iteration proceeds in our experiments, the following experimental results have been found:

1. The estimated solution payoff can reach the true solution payoff closer, which enhances the accuracy of equilibrium results.
2. The game solution has moved from a weak to a strict equilibrium, which improves the quality of game equilibrium.
3. The estimated payoffs for both players increase, which provides a better win–win situation for the game.
4. The asymptotic and marginal stable profiles with respect to the game equilibrium are found to increase, which enhances the game stability.

11.6 Conclusions and Future Directions

In this chapter, we proposed a novel hybrid simulation model which integrates ABS for consumer market activities and SD model for duopoly companies' supply chain operations. Based on the proposed model, we developed a novel GSA procedure, which involves various components such as strategy refinement, data sampling, game solving, and performance evaluation to resolve the simulation-based empirical game. Then, experiments are conducted, where soft drink duopoly scenarios are considered involving different decision variables and experimental iterations. Experiment results have successfully demonstrated:

- Effectiveness of proposed simulation framework in terms of integrating supply chain operations, marketing activities, and estimating the player strategic movement;
- Effectiveness of proposed GSA procedure in terms of achieving reduced estimated errors, improvement, robustness, and stability for game equilibriums.

Future researches will focus on the following aspects. A variety of simulation scenarios are in the list to further test the scalability issues of the proposed simulation testbed with the GSA procedure. A mathematical proof for the effectiveness and convergence of the proposed GSA procedure will enhance the practicability and help to adapt the approach depending on distinct conditions.

References

Adjali I, Dias B, Hurling R (2005) Agent based modeling of consumer behavior. In: Proceedings of the 2005 North American association for computational social and organizational science annual conference, University of Notre Dame, Notre Dame

Albert R, Barabasi A (2002) Statistical mechanics of complex networks. Rev Mod Phys 74:47–97

Asano M, Iryo T, Kuwahara M (2010) Microscopic pedestrian simulation model combined with a tactical model for route choice behavior. Transp Res C 18:842–855

Bertrand J (1883) Book review of theorie mathematique de la richesse sociale and of recherches sur les principles mathematiques de la theorie des richesses. J Savants 67:499–508

Collins J, Arunachalam R, Sadeh N, Eriksson J, Finne N, Janson S (2004) The supply chain management game for the 2005 trading agent competition. Carnegie Mellon University, Pittsburgh

CMO Council CMO Council State of Marketing, Intentions and Investments (2010) http://www.deloitte.com/assets/Dcom-UnitedStates/Local%20Assets/Documents/us_consulting_CMOCouncil_050510.pdf Accessed 25 Nov 2012

Cournot competition (2012) http://en.wikipedia.org/wiki/Cournot_competition. Accessed 23 June 2012

Esmaeili M, Aryanezhad M, Zeephongsekul P (2009) A game theory approach in seller-buyer supply chain. Eur J Oper Res 195:442–448

Hong I, Hsu H, Wu Y, Yeh C (2008) Pricing decision and lead time setting in a duopoly semiconductor industry. In: Proceedings of the winter simulation conference (WSC), pp 2229–2236

Jager W, Janssen MA, Vlek CAJ (1999) Consumats in a commons dilemma: testing the behavioral rules of simulated consumers. COV Report No. 99–01, Centre for Environment and Traffic Psychology, University of Groningen

Jordan PR, Vorobeychik Y, Wellman MP (2008) Searching for approximate equilibria in empirical games. In: Proceedings of the 7th international conference on autonomous agents and multiagent systems (AAMAS), pp 1063–1070

Jordan PR, Schvartzman LJ, Wellman MP (2010) Strategy exploration in empirical games. In: Proceedings of the 9th international conference on autonomous agents and multiagent systems (AAMAS), pp 1131–1138

Kiekintveld C, Wellman MP, Singh S (2006) Empirical game-theoretic analysis of chaturanga. In: Proceedings of the AAMAS-06 workshop on game-theoretic and decision-theoretic agents

Kotler P, Keller KL (2007) A framework for marketing management, 3rd edn. Pearson Prentice Hall, Upper Saddler River

Mckelvey SM (2006) Coca-Cola vs. PepsiCo—A super battleground for the cola wars? Sport Mark Q 15:114–123

Min H, Zhou G (2002) Supply chain modeling: past, present and future. Comput Ind Eng 43:231–249

Mockus J (2010) On simulation of optimal strategies and Nash equilibrium in the financial market context. J Glob Optim 48:129–143

Naik PA, Raman K, Winer RS (2005) Planning marketing-mix strategies in the presence of interaction effects. Mark Sci 24:25–34

North MJ, Macal CM, St Aubin J, Thimmapuram P, Bragen M, Hahn J, Karr J, Brigham N, Lacy ME, Hampton D (2010) Multiscale agent-based consumer market modeling. Complex 15:37–47

Parlar M, Wang Q (1994) Discounting decisions in a supplier-buyer relationship with a linear buyer's demand. IIE Trans 26:34–41

Poropudas J, Virtanen K (2010) Game-theoretic validation and analysis of air combat simulation models. IEEE Trans Syst Man Cybern A 40:1057–1070

Reeves DM (2005) Generating trading agent strategies: analytic and empirical methods for infinite and large games. Ph.D. Dissertation, University of Michigan

Song F, Jing Z (2010) The analysis and system simulation of price competition in a duopoly market. In: Proceedings of Management and Service Science (MASS), pp 1–4

Sterman JD (2000) Business dynamics, systems thinking and modeling for a complex world. McGraw-Hill, New York

Sureka A, Wurman PR (2005) Using Tabu best-response search to find pure strategy Nash equilibria in normal form games. In: Proceedings of the 4th international joint conference on autonomous agents and multiagent systems, pp 1023–1029

Szidarovszky F, Bahill AT (1998) Linear systems theory, 2nd edn. CRC Press, Boca Raton

Unsal HH, Taylor J (2011) Modeling inter-firm dependency: game theoretic simulation to examine the holdup problem in project networks. J Constrn Eng Manag 137:284–293

Venkateswaran J, Son YJ (2007) Effect of information update frequency on the stability of production-inventory control systems. Int J Prod Econ 106:171–190

Vorobeychik Y (2008) Mechanism design and analysis using simulation-based game models. Ph.D. Dissertation, University of Michigan

Vorobeychik Y (2010) Probabilistic analysis of simulation-based games. ACM Trans Model Comput Simul 20(3):Article 16

Vorobeychik Y, Wellman MP, Singh S (2007) Learning payoff functions in infinite games. Mach Learn 67:145–168

Walsh WE, Parkes DC, Das R (2003) Choosing samples to compute heuristic-strategy Nash equilibrium. In: Proceedings of the AAMAS workshop on agent mediated electronic commerce (AMEC V) Melboune, Australia

Wang Q, Wu Z (2000) Improving a supplier's quantity discount gain from many different buyers. IIE Trans 32:1071–1079

Wellman M P (2006) Methods for empirical game-theoretic analysis. In: AAAI'06 Proceedings of the 21st national conference on artificial intelligence, vol 2, pp 1552–1555

Yoshida T, Hasegawa M, Gotoh T, Iguchi H, Sugioka K, Ikeda K (2007) Consumer behavior modeling based on social psychology and complex networks. In: E-Commerce technology and the 4th ieee international conference on enterprise computing, E-Commerce and E-Services, pp 493–494

Yu Y, Liang L, Huang G (2006) Leader-follower game in vendor-managed inventory system with limited production capacity considering wholesale and retail prices. Int J Log Res Appl 9:335–350

Zhang X, Huang G (2010) Game-theoretic approach to simultaneous configuration of platform products and supply chains with one manufacturing firm and multiple cooperative suppliers. Int J Prod Econ 124:121–136

Zhang T, Zhang D (2007) Agent-based simulation of consumer purchase decision-making and the decoy effect. J Bus Res 60:912–922

Chapter 12
Integrating Vendor Managed Inventory and Cooperative Game Theory to Effectively Manage Supply Networks

M. Mateo and E. H. Aghezzaf

Abstract This chapter discusses the issue of integrating inventory and distribution optimization together with game theory to effectively manage supply networks. Inventory and distribution simultaneously optimization is a challenging problem aiming at coordinating decisions related to inventory management with those related to transportation scheduling. This problem is known as the inventory routing problem (IRP) and is an underlying optimization model for supply networks implementing a vendor managed inventory (VMI) strategy. Game theory, and in particular cooperative games, involves several decision-makers willing to coordinate their strategies and share the payoff. In particular, coalitions of decision-makers can make binding agreements about joint strategies, pool their individual payoffs, and redistribute the total in some specified way. In a supply and distribution context, the manager of a franchising business must decide how much inventory to carry. Naturally, the manager of each sales-points wishes to carry an as low as possible amount of inventory and at the same time have enough inventory to cover all demand and not miss any potential sale. One possibility to achieve these two contrasting goals is to allow cooperation among the sales-points and trade the product at some fair price. Sales-points with an excess inventory may want to sell that surplus to other sales-points in the same cluster or coalition, facing a larger than expected demand. The game consists in determining clusters of sales-points which are willing to cooperate, a fair trade-price, and inventory quantities to be carried by each sales-points to minimize the total costs and maximize the total sales. In other models, the total cost of transportation between a depot and a set of customers must be divided among them and the game considers the synergies in the determination of the individual costs.

M. Mateo
Department of Management, Universitat Politècnica de Catalunya (UPC),
Avda. Diagonal 647, p7 08028 Barcelona, Spain

E. H. Aghezzaf (✉)
Department of Industrial Management (EA18), Faculty of Engineering and Architecture,
Ghent University, Technologiepark Zwijnaarde 903 B-9052 Zwijnaarde, Belgium
e-mail: ElHoussaine.Aghezzaf@UGent.be

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_12,

Keywords Supply chain optimization · Distribution systems · Vendor managed inventory · Game theory · Inventory routing problem

12.1 Introduction

Supply chain management is concerned with efficient and effective integration of suppliers, manufacturers, warehouses, and retailers, so that an improved global performance of the supply network can be achieved. In this chapter, we discuss some questions related to the integration of inventory management and distribution together with cooperative game theory. This challenging issue aims at coordinating decisions related to inventory management with those related to transportation scheduling while allowing cooperation among the various nodes of the supply network. Vendor managed inventory (VMI) is one of the effective strategies used to coordinate inventory and distribution. The inventory routing problem (IRP) is an underlying optimization model for the VMI. It involves the integration and coordination of the inventory management and the vehicle routing functions in the supply network. The IRP involves the distribution of some product from a given facility to a set of sales-points, warehouses, stores, etc., over a given planning horizon. We assume throughout this chapter that each sales-point uses or consumes the product at some given rate, which may or may not be constant. Each sales point can maintain a limited amount of inventory of the product, called here local inventory. A fleet of vehicles of same or different capacities are available for the distribution of the product. A typical main objective is the minimization of the total distribution cost during the planning horizon without causing a stockout at none of the sales points.

Game theory can be viewed as an interactive multi-agent decision-making approach. It deals with situations where decision-makers with different competing goals try to take into account others' actions in deciding on the optimal course of action. Game theory is divided into two branches, non-cooperative and cooperative games. In this integrated inventory and distribution management context, we consider cooperative games. We assume that some of the involved decision-makers are willing to cooperate and redistribute the resulting total profit or part of it among them. In a franchising business, for example, managers of the involved sales points must decide how much local inventory to carry. In this typical context, each manager wishes to carry an as low as possible amount of local inventory and is, at the same time, reluctant to miss any potential sale. Therefore, a possible strategy to achieve this goal is to introduce cooperation among similar sales-points. They can trade the product among them at some fair price. Thus, a sales point with an excess inventory may want to sell that surplus to other sales points, in the same cluster, experiencing a larger than expected demand. The game consists in determining clusters of sales points which are willing to cooperate, a fair trade-price, and inventory quantities to be carried by each sales points so that the total cost is minimized and the total sales is maximized. The solution approach consists of developing an inventory and routing planning which allows each sales point to be optimally replenished and at the same time building clusters of sales points willing to cooperate to assure that a smaller possible amount of potential demand is lost.

12.2 Integrated Supply Chain and Related Optimization Problems

Over the past decades, supply chain management (SCM) has been an area of constant progress. A basic idea behind the concept of supply chain or network is a system of facilities and activities that work together, in an effective way, to acquire, produce, and distribute goods to final customers or users. SCM is then basically a set of approaches utilized to efficiently integrate suppliers, manufacturers, warehouses, and retailers, so that merchandise is produced and distributed in the right quantities, to the right locations, and at the right time, while minimizing the system-wide total cost (or maximizing total profit) and satisfying service level requirements (see Shen 2007; Simchi-Levi et al. 2000). The overall goal of SCM is to integrate organizational units and coordinate flows of material, information, and money so that the competitiveness of the supply chain is improved (Stadtler and Kilger 2005). Such integration of material and information flows at various stages of the supply chain becomes more and more conspicuous in the implemented managerial strategies. This integration perspective shifts traditional relationships among the various supply network components from loosely linked independent production units toward a more managerially coordinated network of collaborating production units, to improve the overall efficiency and assure continuous improvement (Bowersox et al. 2002).

The supply chain provides thus a framework within which logistics strategies are to be developed and executed. Logistics includes all activities related to the movement of material and information between the various units of a supply chain. Logistics can essentially be viewed as a planning approach that seeks to create a single global plan for the flow of product and information throughout a supply network (Christopher 1998). SCM then builds upon this global planning approach and seeks to achieve the coordinations between these logistical processes. An important goal in logistics and SCM is to eliminate or reduce to its minimal level the inventory at and between suppliers and customers, through information sharing on demands and current on-hand stock levels for example. A possible strategy to achieve this goal, for example, is allowing the supplier to take responsibility for the replenishment of the customers' inventory instead of waiting for the customers to place orders. Thus, on the basis of information related to their sales, their current inventory levels, details of any other marketing activities, and so on, the supplier replenishes their inventory accordingly. This is the well-known concept of

"Vendor Managed Inventory" (VMI) (for more details on VMI see for example Christopher 1998; Waller et al. 1999). VMI is an agreement between a vendor and his customers according to which customers agree to the fact that the vendor decides the timing and size of the deliveries. Compared with the traditional nonintegrated inventory replenishments and distribution plans, overall inventory, and distribution performances throughout the supply chain are by far superior when VMI is implemented. VMI is thus a win–win partnership for both the vendors or suppliers and their customers. The advantage to the customers is that the risk of stockouts reduces drastically, while they do not have to make great efforts on inventory management. As a consequence, this decreases significantly their inventory related costs. On the other hand, vendors can realize significant savings on distribution-related costs by being able to better plan and schedule product deliveries, as a result of a direct access to detailed information on actual product usage by their customers.

Under the VMI policy, the complex task of developing a distribution plan, which guaranties that the customers are never in stockout while minimizing at the same time the total inventory costs and realize potential savings in distribution costs, arises. Thus, decisions of both inventory management and transportation scheduling are to be integrated. Since the integrated distribution plan consists of controlling inventory and developing optimal routes for the transporters, this problem is known as "Inventory Routing Problem" (IRP) in the literature. The IRP is one of the challenging optimization problems in the design and management of supply and distribution networks. It also provides a very good starting point for investigating the integration of different components in logistical and supply networks, for instance technology and planning integration, internal integration, material and service supplier integration, which are traditionally dealt with separately. Such integrations are expected to lead to some important improvements in logistics and SCM.

In general, the main objective in the IRP is to design vehicle routes and determine delivery quantities that minimize transportation costs while controlling inventory costs. Dror and Ball (1987) defined this problem in the following way: "the IRP is a distribution problem in which each customer maintains a local inventory of a product such as heating oil or methane and consumes a certain amount of that product each day. Given a central supplier, the objective is to minimize the annual delivery costs while attempting to insure that no customer runs out of the commodity at any time". Clearly for an IRP, the practitioners have to make decisions simultaneously on the following three aspects: the time and frequency of replenishment, the quantity to be delivered to each customer, and the delivery routes. The first two problems are related to inventory control, whereas the last one corresponds to routing and distribution. As a consequence, the IRP can be seen as an extension of the well-known vehicle routing problem (VRP). Often the context of the VRP is that of using a fleet of vehicles, with limited capacity, to deliver goods located at a central depot to customers who have placed orders for such goods. The objective is usually to minimize the cost of distributing the goods, while satisfying all possible side-constraints that are imposed on the problem.

One can see that the IRP differs from traditional vehicle routing problem because it is based on customers' usage of the product rather than customers' orders (Campbell et al. 1998; Tempelmeier 2011). Recently, a thorough research has been conducted in the inventory routing problem. Many variants of the IRP can be found in the literature, ranging from deterministic to stochastic models. In this chapter, the IRP is studied as an underlying optimization model for the VMI policy. In particular, the case when customer-demand rates are assumed to be known and relatively stable and the developed distribution plan can be executed on a periodic basis. This problem is called the cyclic inventory routing problem (CIRP). To gain a better understanding of the general CIRP, the subsequent section provides a review and an introduction to this model, and discusses a column generation-based heuristic which can be used to solve the problem.

12.3 The Inventory Routing Problem

The Inventory Routing Problem (IRP) is typically concerned with the repeated distribution of a single product (SKU, i.e., stock-keeping unit), from a single facility to a set of customers over a given planning horizon, that can possibly be infinite. The customers consume the product at a given rate and have the capability to maintain a local inventory of it. A fleet of vehicles of same or different transportation capacities is available for the distribution of the product. The objective is to minimize the total distribution and inventory costs during the whole planning horizon without causing any stockout at none of the customers.

As already mentioned, the IRP can be seen as an underlying optimization model for the VMI strategy involving integration and coordination of the inventory management and vehicle routing components in a logistical chain. Over the past decades, various models for IRP have been proposed and studied by different researchers. Noticeably, the IRP defined above is deterministic due to the fact that the customers' consumption rates are assumed to be known and relatively stable. However, if we consider the issue of coordinating inventory control and transportation scheduling from a practical point of view, stochastic models might better describe many real-life cases. Actually, the literature of IRPs tends to classify the IRP models according to four key characteristics, as introduced below (see for example Shen 2007):

- length of the planning horizon, which may be either finite or infinite;
- demands, which can be either deterministic or stochastic;
- fleet size, i.e., the number of available vehicles, which is either limited or unlimited;
- number of customers visited on a vehicle trip, which some models limit to be one whereas others allow multiple customers on a single route.

Both Kleywegt et al. (2002) and Adelman (2004) offer an excellent classification of the IRP models using the above four categories. More recently, Andersson et al. (2010) carried out a detailed description and a comprehensive classification of the IRP models. Beyond the listed above four characteristics that are usually used to study similarities and differences between different IRP models, the authors also have introduced extra dimensions to reflect the focus on industrial aspects, such as the topology of the problem, in which three modes may be discussed: one-to-one, one-to-many, and many-to-many. In particular, they classified the regular IRP models in the literature according to the length of the planning horizon into instant, finite, and infinite planning horizon problems.

The cyclic inventory routing problem (CIRP) belongs to the class of infinite planning horizon IRP. For this class of problems the appropriate objective function to be minimized is the long-run average distribution and inventory management costs. In brief, this particular IRP considers a special case, in which a single distribution center, supplying a single product, serves a set of customers implementing economic order quantity-like policies to manage their inventories. Each customer is served by an assigned vehicle in a cyclic manner and in such a way that at no moment a stockout should occur at the customer. Demand rates at the customers are assumed to be stable and their averages are known to the supplier. The objective is to minimize total fleet operating, inventory holding, and distribution costs over a determined planning horizon. In this section, we review the model for this CIRP, proposed in Aghezzaf et al. (2006), as well as the model for its resulting subproblem when using a column generation-based approach to solve the CIRP. This model is a variant of the general IRP problem, where distribution is planned, in a cyclical way, from multiple distribution centers, denoted by r_k ($k = 1,\ldots, p$) to a set of sales points S ($i = 1,\ldots, n$). Each sales point i in S has a demand rate d_i, given in units per unit of time. A fleet of homogeneous vehicles V having capacity κ are used to supply these sales points.

12.3.1 The Mixed Integer Formulation of the Problem

In case of a single depot r, the natural mixed integer formulation of the problem proposed in Aghezzaf et al. (2006) is based on the following assumptions:

1. The time necessary for loading and unloading a vehicle is relatively small compared with travel times, and it is therefore neglected.
2. Inventory capacity at the sales points is assumed to be large enough, so corresponding capacity constraints can be omitted from the model.
3. The travel costs are assumed to be proportional to the travel times, so we only use a parameter δ that indicates the cost per hour of travel.

Let t_{ij} (in hours) denote the duration of the trip from sales point $i \in S^+ = S \cup \{r\}$ to sales point $j \in S^+$. Also, let φ_i and η_i denote, respectively, the inventory fixed ordering and holding costs at sales point i. The fixed operating and maintenance costs of vehicle $v \in V$ (in Euro per hour) is denoted by ψ^v. The solution of the problem consists of replenishing sales points assigned to each vehicle $v \in V$ during related cycle time.

The model variables are:

- x_{ij}^{v}: A binary variable assuming 1 if the sales point j is served by the vehicle v immediately after the sales point i; and 0 otherwise.
- y^{v}: A binary variable assuming 1 if the vehicle v is used; and 0 otherwise.
- z_{ij}^{v}: The sum of demand rates (units per hour) of remaining sales points in a tour covered by vehicle v when it travels to a sales point j immediately after it has served sales point i. This quantity equals zero when the trip (i, j) is not on any tour made by vehicle v.
- T^{v}: The cycle time (in hours) of the multi-tour made by vehicle v. It must be strictly positive to avoid division by zero in the objective function.

The initial nonlinear mixed integer formulation of the IRP:
IRP_{IP}: Minimize

$$Z = \sum_{v \in V} \left[\begin{array}{l} \left(\psi^{v} y^{v} + \frac{1}{T^{v}} \left(\sum_{i \in S^{+}} \sum_{j \in S^{+}} \delta t_{ij} x_{ij}^{v} \right) \right) + \\ \left(\sum_{i \in S} \left(\varphi_{i} \frac{1}{T^{v}} + \frac{1}{2} \eta_{i} d_{i} T^{v} \right) \left(\sum_{j \in S^{+}} x_{ij}^{v} \right) \right) \end{array} \right],$$

Subject to:

$$\sum_{v \in V} \sum_{i \in S^{+}} x_{ij}^{v} = 1, \quad \forall j \in S \tag{12.1}$$

$$\sum_{i \in S^{+}} x_{ij}^{v} - \sum_{k \in S^{+}} x_{jk}^{v} = 0, \quad \forall j \in S,\ v \in V \tag{12.2}$$

$$\sum_{i \in S^{+}} \sum_{j \in S^{+}} t_{ij}^{v} x_{ij}^{v} - T^{v} \leq 0, \quad \forall v \in V \tag{12.3}$$

$$\sum_{v \in V} \left(\sum_{i \in S^{+}} z_{ij}^{v} - \sum_{k \in S^{+}} z_{jk}^{v} \right) = d_{j}, \quad \forall j \in S \tag{12.4}$$

$$x_{rj}^{v} - y^{v} \leq 0, \quad \forall v \in V,\ j \in S \tag{12.5}$$

$$T^{v} z_{rj}^{v} \leq \kappa, \quad \forall j \in S,\ v \in V \tag{12.6}$$

$$z_{ij}^{v} - \left(\sum_{k \in S} d_{k} \right) x_{ij}^{v} \leq 0, \quad \forall v \in V,\ i,j \in S^{+} \tag{12.7}$$

The constraints (12.1) make sure that each sales point is served by one and only one vehicle. Constraints (12.2) assure that a vehicle assigned to serve a sales point will serve this point in and will leave to a next sales point. The constraints (12.3) indicate that the cycle time of a vehicle should be greater than or equal to the total time it has to travel. It includes the duration of all tours made by the vehicle. The constraints (12.4) assure that the cumulated demand rates, per hour, of the remaining sales points in the tour carried by the vehicle $v \in V$ serving the sales point $j \in S$ is reduced by the demand rate d_j when the vehicle leaves this sales point. Constraints (12.5) make sure that a vehicle can only leave the distribution depot r to make a voyage if it is being used. The constraints (12.6), which are nonlinear, are vehicle capacity restrictions. Constraints (12.7), finally, assure that z_{ij}^v cannot carry any cumulated demand rates unless x_{ij}^v equals 1.

12.3.2 Possible Solution Approaches

In the following, we summarize the approximation algorithm which was proposed to solve this model (see Aghezzaf et al. 2006). The obtained solution forms the basis for the clustering of the sales points. Cooperation is then brought in each cluster to minimize the costs related to inventory and allow for an efficient management of the total inventory at the cluster's level. The approximation algorithm is a column generation framework in which the subproblem generating the vehicle multi-tours is solved heuristically. More specifically, the subproblem is solved using a heuristic which extends the savings-based heuristic used to solve the vehicle routing problem.

A multi-tour v is defined by the binary matrix X^v, the matrix Z^v, and the parameter T^v such that $x_{ij}^v = 1$ if multi-tour v contains a trip that visits the sales point j immediately after it has visited the sales point i. The cost rate of the multi-tour v can be computed as follows:

$$K^v = \psi^v + \frac{1}{T^v}\left(\sum_{i \in S^+}\sum_{j \in S^+} \delta t_{ij} x_{ij}^v\right) + \sum_{i \in S}\left(\varphi_i \frac{1}{T^v} + \frac{1}{2}\eta_i d_i T^v\right)\left(\sum_{j \in S^+} x_{ij}^v\right)$$

Let PV be the collection of all possible multi-tours described by (X^v, Z^v, T^v) satisfying the required restrictions, then IRP can be reformulated as follows (called as IRP_{MG}):

$$\text{IRP}_{\text{MG}}: \qquad \text{Minimize} \sum_{v \in PV} K^v\, w_v \tag{12.8}$$

Subject to:

$$\begin{aligned} &\sum_{v \in PV}\left(\sum_{i \in S+} x_{i,j}^v\right) w_v = 1, \quad \text{for all } j \in S \\ &w_v \in \{0, 1\}, \qquad \text{for all } v \in \text{PV} \end{aligned} \tag{12.9}$$

This new problem seeks the best combination of multi-tours ($w_v = 1$ if the multi-tour v is selected, and 0 otherwise) which covers the sales points while minimizing the total cost. This problem $\mathrm{IRP_{MG}}$ is the master problem in the column generation process. As the number of possible columns is typically huge, an efficient procedure of generating columns is necessary to determine the set of suitable multi-tours to consider in the problem $\mathrm{IRP_{MG}}$ so that it reaches a near optimal solution as fast as possible. This multi-tour generation, called procedure P1, is summarized below:

1. Initially, a list of multi-tours PV is created with the $|S|$ 'basic' multi-tours, i.e., those visiting one and only one sales point.
2. The linear programming relaxation of problem $\mathrm{IRP_{MG}}$ is solved.
3. From the solution of the LP-relaxation of problem $\mathrm{IRP_{MG}}$, dual prices λ_j for each $j \in S$, associated with constraints (12.9), are generated.
4. An attempt to determine a new feasible multi-tour with a negative reduced cost is launched. If such a multi-tour is found, it is added to the list PV and the process returns to step 2. If there is no such multi-tour, the LP optimal solution is reached and the process moves to step 5.
5. Problem $\mathrm{IRP_{MG}}$ is solved (PV is the set of all feasible multi-tours which are generated) to provide the final solution.

To determine the feasible multi-tour with the least reduced cost, a multi-tour-generator subproblem IRPSP must be solved. This can be done with an extended savings-based heuristics (for more details see Aghezzaf et al. 2006, or Raa and Aghezzaf 2009). Below we provide some information of this subproblem known as single vehicle cyclic inventory routing problem (SV-CIRP).

Another approximate model for the IRP is a multi-period inventory routing problem (MP-IRP) where the customers are assumed to consume the product at deterministic and constant rates. The MP-IRP considered here is concerned with a distribution system using a fleet of vehicles to distribute a product from a single depot to a set of customers having deterministic and constant demands. The distribution policy is executed over a given finite horizon, for example on a set of consecutive periods (or days). The objective is then to determine the quantities to be delivered to the customers, the delivery time, and to design the vehicle delivery routes, so that the total distribution and inventory costs are minimized. Here also, the resulting distribution plan must prevent stockouts from occurring at all customers during the given planning horizon.

A possible solution approach for this problem is the use of Lagrangian relaxation. Lagrangian relaxation was developed in the early 1970s with pioneering work of Held and Karp (1970, 1971) on the traveling salesman problem. It is now one of the most prevalent techniques for generating lower bounds to use in algorithms to solve combinatorial optimization problems. Lagrangian relaxation involves attributing Lagrangian multipliers to some of the constraints in the formulation of the problem and relaxing these constraints into the objective function, and then solving the resulting subproblems to determine a lower bound on the optimal solution to the original problem.

In a possible implementation of Lagrangian relaxation for the MP-IRP, the delivered load constraints can be assumed to be the complicating constraints. Along the same lines as in Zhong and Aghezzaf (2012), the relaxed problem decomposes into an inventory allocation subproblem and a vehicle routing subproblem. These subproblems involve fewer variables and constraints, respectively, and they can be solved more efficiently than the original problem. The structure of this Lagrangian relaxation approach is shown in Fig. 12.1. A thorough discussion of this approach can be found in Zhong and Aghezzaf (2012).

12.4 Game Theory and Supply Chains

Borm et al. (2001) surveyed the research conducted on cooperative games associated with several operations research problems involving various decision-makers. These decision-makers, also considered as cooperating players, face together a common optimization problem for which they seek a solution that minimizes some total joint cost. Afterwards, once the solution for this cost optimization problem is obtained, the decision-makers face an additional supplementary problem related to the allocation of the resulting total cost. This allocation consists of determining how to distribute this joint total cost among the involved individual players. The problems tackled are classified according to their nature. Routing and inventory are among the treated problems. However, they are not considered simultaneously as is the case in this chapter.

Game theory can also be adapted to the distribution problems within the context of the supply chains. The usual design of an inventory system is hierarchical, because the transportation flows connect an element in an echelon with another in the next echelon of the supply chain. In a three-echelon supply chain, flows can stream from manufacturers to wholesalers and then from wholesalers to retailers. The flows between elements of the same echelon provide more flexibility to the supply chain or network. The members in the same echelon of the supply chain can pool their inventories in order to achieve a good balance or trade-off between low inventory levels and good service levels. An obvious alternative to avoid losing sales opportunities is to hold an important amount of stock to cover demand peaks, however this comes with an overstocking cost. Nevertheless, these overstocking costs can be overcome through the use of game theory.

The cooperation mechanisms can be developed at different levels, according to the targeted coverage of the demand peaks. If a demand increase is experienced only in few contained sales points of some area, the cooperation between the sales points in that same area can take place. However, if the demand increase is experienced in a large geographic area, all the sales points will experience the same demand increase and the cooperation will become difficult if not possible. In this case, an alternative would be to define the cooperation game rather at the level

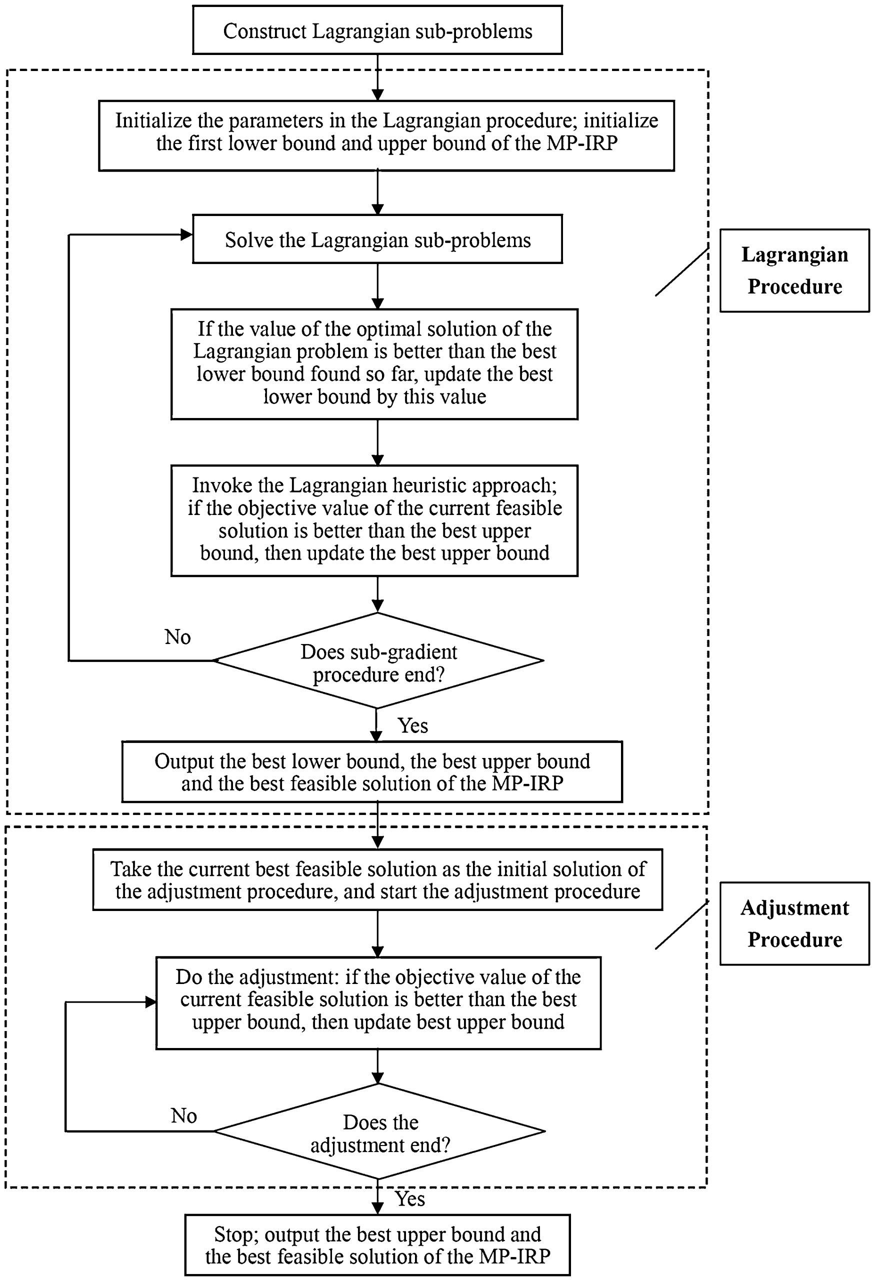


Fig. 12.1 Overview of the Lagrangian relaxation approach

and between distribution warehouses. Therefore, the cooperation that is intended for solving local inventory management problems can be extended to similar situations considering elements of the same level in the supply chain.

Mateo et al. (2012) integrated game theory and a distribution problem to maximize the sales opportunities of the company under variables demand rates. The basic idea is the creation of cooperating cells. A typical cooperating cell includes sales points which are located within some delimited geographic area and are willing to cooperate. Thus, different groups of sales points will be created over the territory and the cooperation within each group of sales points can be seen as a local cooperative game. In fact, this leads to a practical and realistic solution. When a sales point has not enough stock available, extra inventory of the closest sales points could be used as a solution to ensure as shorter as possible stockout duration. In a franchising company, this fact is very coherent with the franchising culture, which aims at establishing some cooperation between its franchisee sales points. Arshinder et al. (2008) showed that when a franchising company takes the responsibility for managing inventory of its sales points, several advantages are observed. In particular, inventory levels decrease and response times to demand decrease.

Cooperative game theory also studies games in which selfish players collaborate to increase their benefits. Özener (2008) considered the cooperative game consisting in a replenishment cost allocation problem of the VMI. As VMI replenishment is collaboration between a supplier and its customers, the problem is how allocate the distribution costs incurred by the supplier among its customers. The simplest methods ignore relations between the customers, due to their locations, usage rates, and storage capacities. As a result, the price charged to a customer for distribution is not representing the actual cost of that customer's service. Although the customers do not form coalitions in this VMI problem, the mechanism to compute the cost-to-serve for each customer is equivalent to a cost allocation method for the Inventory Routing Game (IRG). The customers represent the players and the optimal objective function value of the IRP is the characteristic function value of the game. Cost allocation in routing and distribution problems has rarely been present in the literature. Up to Özener (2008), no published work had addressed the issue of allocating cost of an IRP. Nevertheless, there are closely related analyses on the traveling salesman game (TSG) and the vehicle routing game (VRG).

The TSG considers the problem of allocating the cost of a round trip among the cities visited. Depending on the game, a home city may not be included in the set of players. The VRG considers the problem of allocating the cost of optimal vehicle routes among the customers served. Cost allocation methods studied in the literature for these related problems, such as TSG and VRG, do not have to deal with the time or inventory components of the problem and hence cannot be applied directly to IRG.

The benefits of collaborative decision-making in a supply chain have also been extensively studied. Lau et al. (2008) argued that a better coordination of supply chains (distribution can be considered a part of the chain) is achieved with an appropriate choice of inventory management policies. The higher the level of information sharing, the more efficient the supply chain performs. Zhang and Zhang (2007) used a simulated supply chain to quantify the profit generated by sharing information on demand. Wu and Cheng (2008) defined three levels of information sharing to implement VMI and improve efficiency in logistics chains. Southard and Swenseth (2008) examined the effectiveness of VMI in various non-traditional contexts, and found that sufficient economic benefits could be achieved with the use of a technology-enabled VMI.

In an inventory system, lateral transshipments are stock movements between locations of the same echelon (Paterson et al. 2011). The transshipments can be held periodically at predetermined times to proactively redistribute stock or to reactively meet the unsatisfied demand from the stock on hand. Several models for different systems have been considered. Hu et al. (2005) developed a model for several stocking locations, which can be used to approximate ordering policies. They conclude that transshipments are effective to reduce inventories if the transshipment cost is smaller than the holding and stockout costs. The model assumes that transshipments are free and instantaneous. Kukreja and Schmidt (2005) extends this model for compound Poisson demand processes and (s, S) replenishment policies, for which they propose a simulation-based method. The most common works are based on the optimal transshipment policy between a supplier and a set of retailers. For instance, Minner and Silver (2005) show an application where an (s, q) policy is used for replenishing stock. Whenever the inventory reaches or falls below the reorder point s, a batch of size q is ordered from the depot or the supplier.

Transshipments are used in the literature in response to stockouts in one-for-one replenishment policies (Grahovac and Chakravarty 2001), continuous review replenishment policies (Minner et al. 2003; Kukreja and Schmidt 2005) and periodic review replenishment policies (Zhao et al. 2006; Archibald et al. 2010). Models of retail networks with periodic replenishment often assume simultaneous replenishment of all locations (Cao and Silver 2005; Herer et al. 2006; Archibald et al. 2009). However, in Zhao et al. (2006) a location has no information about the inventory level at other locations in the network.

Concerning to the proactive lateral transshipments, Paterson et al. (2011) differed two cases: the standalone redistribution, without replenishments, and the redistribution combined with replenishments. In the standalone redistribution, Banerjee et al. (2003) and Burton and Banerjee (2005) compared the performance of a proactive redistribution policy and a simple reactive transshipment method. The supply at each location is leveled with a redistribution policy called transshipment inventory equalization (TIE). On the other side, in a reactive policy called transshipment based on availability (TBA) transshipment is done in case of shortages and prevents a stockout. Both studies underline how the overall objectives can influence the suitability of transshipment policies. They assume that replenishment orders are placed according to a periodic base-stock policy, where the period is set equal to 20 days and the order-up-to level is equal to the average demand during lead time plus review period.

If redistribution is combined with replenishments, the optimal period for redistribution is difficult to determine. For instance, Tagaras and Vlachos (2002) demonstrated that the redistribution benefits have a great influence on the performance of the policy when demand is highly variable. They specially focus on the sensitivity of the policy based on the variability within the demand distribution. They consider non-negligible transshipment times in a two location system. The redistribution point using simulation is optimized and they conclude that preventive transshipments are generally beneficial.

About the reactive lateral transshipments, Paterson et al. (2011) considered centralized and decentralized systems. In a two echelon centralized system several ways can be performed to satisfy stockouts through emergency stock movements. One of them is the lateral transshipments, but sometimes emergency shipments from the central warehouse are also valid. Wee and Dada (2005) worked on this problem with several combinations of transshipments and proposed a method for deciding which emergency stock movement is optimal under a set of circumstances.

In a decentralized system, each stocking point operates to meet its own goals. Slikker et al. (2005) studied when independent vendors benefit by cooperating as a grand coalition. They modeled the problem as a general newsvendor situation with N retailers. It is shown that retailers' cooperation may always achieve a higher profit, using a game theoretic approach, as no retailer has an incentive to leave the grand coalition. In this study, transshipment costs are not included.

Rudi et al. (2001) considered how cooperation can be established in a newsvendor type model with a manufacturer and two retailers. Transshipment prices are determined in advance by an accepted authority, for example by the manufacturer. Rudi et al. (2001) showed that there exists a Nash equilibrium for the ordering quantities, and that the joint profit is generally not maximized at this equilibrium. Hu et al. (2007) focused on dealing with linear transshipment costs. An extension of this model to N retailers, where the price is based on the dual of the transshipment problem, is worked by Anupindi et al. (2001). It is shown that this rule is always in the core of the corresponding transshipment game. Game theory can be adapted to this distribution problem in order to maximize the sales opportunities of the company under varying demand rates and to increase its revenues (Chan and Lee 2005; Ghiani et al. 2004). When a subset of players, which is called a coalition of players, use a mechanism to work together in order to reduce their costs, the problem turns into an IRG.

Özener (2008) defines the IRG as a cooperative game where the supplier has to serve the n customers that are the players in the game. The set of all the customers N is the grand coalition of the game and any subset $S \subset N$ of the customer becomes a coalition. The characteristic function $c(S)$ is the optimal average transportation cost of the IRP with a given set of customers S, also members of a cooperative cell, over the planning horizon and $c(N)$ is the total cost of the grand coalition. He assumes that the number of trucks to serve any subset of the customers is enough. The cost allocations for the IRG represent the cost-to-serve values for the customers.

As mentioned above, Slikker et al. (2005) deal with the problem of a general supplier situation with m retailers and analyze the benefit of independent vendors from cooperating. Using a game theoretic approach, it is shown that if retailers cooperate they can achieve a higher profit. No retailer has an incentive to leave the coalition. They clearly illustrate that centralized ordering and transshipments can also be positive for decentralized systems. In a similar way, Hu et al. (2007) consider a model with one manufacturer and two retailers. They establish some conditions derived for the existence of transshipment costs which induce retailers to make jointly optimal decisions, i.e., cooperation.

Two fundamental concepts in cooperative games are the nucleolus and the Shapley value. According to Schmeidler (1969), nucleolus is the value that lexicographically maximizes the minimal gain, the difference between the standalone cost of a subset and its coalition cost, over all the subsets of the collaboration. If the core is non-empty, the nucleolus is included in the core. Nucleolus may still exist when the core is empty when a cooperation cost is balanced and the cost assigned to each player is less than or equal to their standalone costs. Another well-known cost allocation method is the Shapley value, which is defined for each player as the weighted average of the player's marginal contribution to each subset of the collaboration (Shapley 1953).

In IRG, TSG, and VRG, computing the characteristic function value for a given coalition requires solving an NP-Hard problem and calculating generic cost allocations such as the Shapley value is impractical since it requires considering explicitly an exponential number of subsets. Due to the decisions on the delivery volume and the fact that IRP is a multi-period problem, there exist practically an infinite number of feasible distribution patterns and so computing the exact characteristic function value is very challenging even for small instances of the problem.

Finally, Granot and Sosic (2003) showed that even if retailers can decide how much to share, it may happen that no residual inventory is distributed and no additional profit is reached.

12.5 Applications

In this section, we present and discuss two applications using collected data, similar to case studies, such as that of Gaur and Fisher (2004) which considers a periodic IRP at a supermarket chain. We will concentrate on the stream of literature focusing on cooperative game theory in the IRP problem.

12.5.1 Case Study 1: Cooperation in the Wine Distribution

12.5.1.1 General Description

The case discussed in (Mateo et al. 2012) is that of a leading retailer of wine and liquor. Its business model is based on franchising with more than a thousand stores scattered all over the country. These specific stores represent approximately 30 % of the sales. This percentage is subdivided as follows: 14 % is distributed through the sector of retail chains, 10 % is distributed to independent groceries, and the rest is distributed to independent sales points specialized in wine products. When the work was done, the company sold as much as 3.5–4.2 % of the total amount of wine sold in UK. The study analyses the distribution of a single product from the local warehouse of the company to each one of the franchised stores. The special product is distributed to some 116 sales points scattered over some 49 cities. For the distribution of the product a fleet of exclusive vehicles is used. The fleet is composed of three vehicle categories: vans of 1.5 tons, trucks of 3 tons, and trucks of up to 10 tons.

A van of 1.5 tons is usually used to serve sales points with low demand. A truck of 3 tons has a capacity of 4 pallets and is used for sales points at short-distances, with limited load and frequent trips, from the distribution center. Trucks of up to 10 tons compose the largest category in the fleet, with capacity from 6 to 12 pallets. To assign a cost to intermediate trucks (between 3 and 10 tons), a 5 % on the fix cost and a 10 % on the variable cost is discounted. The standard pallet used throughout the paper is the British pallet (1.2 × 1 m). The 1.2 m side is used for the vehicle width (allowing a maximum of two pallets). The capacity in pallets of a truck is determined by its length. This capacity is given by 2 pallets multiplied by the length of the truck in meters). The largest vehicle carries 12 pallets.

12.5.1.2 Theoretic Base

The main parameter is the inventory (in the first application presented will be the number of bottles) required by a 'cooperative cell' to solve possible stockout problems of its m sales points $G = \{1,\ldots, m\}$. Let us denote by f the total number of required bottles. The set of service providers, the sales points in the cooperative cell, are the n players in the 'cooperative game' and is denoted by $N = \{1,\ldots, n\}$. The offer of services of the players (offer profile OPi by each $i \in N$) is the number of bottles that each sales point $i \in N$ is willing to hand over. This value depends on the dimension of the sales point and the forecasted demand. Therefore, there is a cooperative game problem on the demand, which is denoted by $\Omega = (N, G, \mathrm{OP}, f)$, for each group of sales points in N. For this collaborative game, the core may be empty. In a stable behavior, the cost of the cluster should be less than or equal to the cost of an individual behavior.

In order to give to the sales points an incentive to cooperate, the agents come to an agreement that bottles for a collective usage will be sold at higher price in such way that a third part of the total benefit is for the sales point which is handing over the bottle. The profit for each possible coalition is computed sequentially. First, the sales points are considered as isolated ($m = n = 1$), with a nil profit as no bottle can be handed over if the sales point is alone. Then, the coalitions of two sales points are considered ($m = n = 2$), then coalitions of three sales points, and so on.

The procedure that applies the principles of the game theory runs as follows:

1. The total number of bottles which can be supplied by all sales points in the cell is computed.
2. The number of bottles that can be distributed (lower value between the supply and demand) is determined.
3. If the total supply is larger than demand, the demand will be distributed. Otherwise, if the total demand is larger than the supply, the number of bottles is limited to the supplied amount. The Shapley values are considered to be split among the members of the cooperative cell.

Thanks to the existence of the cooperative cell, an additional number of bottles can be sold and generate an extra common profit (Wong et al. 2009). In this case, once heard the opinion of the managers, this extra profit is split by the franchising company in three parts: two-third for the sales points that receives the transhipment and one-third for the supplier sales point (Vinyes 2007). Obviously, other distributions could be taken into account, but they were not tested.

In addition, there is an additional term that is subtracted to this profit. If a seller has a single bottle at the end of a period, he will have a greater fear to put it for cooperation, as he might be losing two-third of the profit in case the possibility of selling this bottle appeared. On the other hand, if a sales point has a larger number of bottles at the end of the same period, the fear to put some or all of them at the community's disposal is smaller as he may think that it might be difficult to sell all these bottles. So, due to this factor, the coalition with sales points assignors and only one bottle in stock will give a low global profit. The value 2.2 in the function expresses this fear and is empirically defined, i.e., from a practical point of view. After testing several values, it shows the practical limit between cooperation and no cooperation. Therefore, the profit function (per period) for coalition N has the following pattern:

$$\upsilon(N) = \left[1 - \sum_{i \in N} \left(\frac{1}{2.2 + \text{Exchange}_i}\right)\right] \text{profit}(N)$$

where profit (N) is the benefit of the total interchanged bottles for coalition N.

Known this coalition profit, the Shapley value of all included sales points is determined. These values indicate the percentage or weight that each sales point has into the global coalition profit and, therefore, they will be the first to give or receive bottles. The Shapley value is an approach to fairly allocate the gain obtained by cooperation among several sales points, being $\varphi_i\ (v)$ the value for sales point i. The amount that each sales point gets is given by Shapley (1953):

$$\varphi_i(v) = \sum_{S \subseteq N - \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} [v(S \cup \{i\}) - v(S)]$$

In that analysis only small groups of sales points are considered. This means coalition of up to four sales points ($m = n = 4;\ 2 \leq |N| \leq 4$). The profit is computed for any possible cooperation of four sales points taken from one up to a single group of four, and the one that maximizes the extra profit is selected.

12.5.1.3 Results of the Cooperative Game

First, the tour design minimizes the distribution cost. The tours are designed using a multi-tour heuristic for the IRP. The objective is to minimize the distribution costs with shorter cycles going through the distribution center and serving the client more frequently. The stock at each sales point is controlled by taking into account the forecasted demand and the frequency of the planned visits. A normal distribution is considered for the demand at each shop. The deviation as high as the clients' requirements are not exactly known. Therefore, a demand per time period is simulated at each shop. Since demands are assumed to follow a normal distribution with a constant average, theoretically after some time all sales points in a cell would have handed over and received the same number of bottles. In order to communicate each one's demands and offers, the time period is decided among the members of the cooperative cell. The study establishes it in twice a day or 4 h, i.e., half labor day.

In the tour design, it was assumed that the demand of sales points is the same and is constant. If it were assumed to be variable, the distribution problem would vary because the tours would differ each time, which sometimes is considered an inconvenience.

The model used for the problem is simplified assuming that it would provide a valid solution in mid and long term. Table 12.1 shows an example of such a simulation. The demand at each sales point is generated from a normal distribution with a mean of five bottles and a standard deviation of 2.5. Once determined, the generated values for the demand and a forecasted stock for each sales point, a bottle excess or shortage is defined for each sales point (Table 12.1), and the Shapley values are generated (Table 12.2). This value gives some indication about the potential extra profit that the sales point contributes for any coalition in which it participates.

In the simulation for a subset of 116 sales points, the additional bottles represent 11 % of higher sales with respect to the quantity without any cooperation. If the results are extrapolated to the whole set of sales points and the number of days per year, the following results are obtained:

Table 12.1 Example of cooperative cell formed by 4 sales points; offers are greater than needs (Mateo et al. 2012)

Sales-point	Demand	Stock	Need	Offer
S1	2	4	0	4
S2	4	2	0	2
S3	7	−1	1	0
S4	5	1	0	1
Total	18	6	1	7

Table 12.2 Shapley values and extra flows in a cooperative cell with offers greater than needs (Mateo et al. 2012)

Sales point	Shapley value	Potential profit (*n*)	Potential profit (*o*)	Extra flow
S1	0.1081	0	0.1621	−1
S2	0.0697	0	0.1045	0
S3	0.6036	0.9055	0	+1
S4	0.0573	0	0.0859	0

- Year sales with cells in the company: 3,825,158.05 €
- Year sales without cells in the company: 3,422,399.43 €
- Extra profit per year: 402,758.62 €.

Clearly, the cooperation between subsets of sales points helps achieve the sales objectives fixed by the company. Therefore, its implementation is recommended.

In order to determine the ideal routes or tours, the algorithm can be used for different kinds of vehicles and their capacities. Thus, known the routes, it is necessary to decide how many vehicles will be used to supply each sales point with the desired frequency. Taking into account these values, the global cost of the distribution on the geographical zone is calculated.

In the paper, only one product is considered. For all sales points, the demand of bottles each 4 h is about 575. The van, the smallest mean of transport, can transport 1,800 bottles. Therefore, the solution is made of tours which can be obtained by some TSP heuristic. All vehicles of the company can supply the demand of the sales points using less than 50 % of their capacity. As the capacity is not an active constraint, the cheapest transportation mean, the van, is considered. If several products were simultaneously considered, the demand would increase and the capacity of vehicles would become critical.

The provided solution has a cycle time of approximately 74 real hours, which are over nine working days. This fact means that to supply the sales points each 4 h a fleet of 19 vehicles are necessary. If the cost of a cycle is 480.45 €, the annual cost is 356,000 €.

12.5.2 Case Study 2: Cost Allocation in the Gas Distribution

12.5.2.1 General Description

Özener (2008) describes another real situation in the context of a large industrial gas company that operates a VMI resupply policy. The company replenishes the tanks at customer locations under the company's control and uses a homogenous fleet of trucks. The customers and trucks are assigned to a facility; trucks and storage tanks contain only one type of product. He seeks to integrate cooperative game theory methods with optimization techniques and develops mechanisms to create stable collaborations. Some earlier works were devoted to collaborative approaches focused on procurement, but he takes the practical facts of the situation into consideration to develop collaborative solutions, ensuring that these mechanisms are viable in that environment. The optimization techniques are useful to the difficult evaluation of synergies in transportation and distribution systems.

The inventory holding costs are ignored because these costs, at the production facilities, and the customer sites are assumed by the company. Therefore, he seeks a feasible distribution plan and is less worried about the cost. The consumption rates of customers are assumed to be deterministic and stationary over time. Briefly, the problem is to minimize the total cost of transportation of a single product from a single depot to a set of customers with deterministic and stationary demand over a planning horizon, with the possibility to be infinite. The objective is to design a mechanism capable of computing a cost-to-serve for each customer that properly accounts for the synergies among customers. Synergies are defined as the interactions among the customers in one of the routes created in the IRP.

A way that allocates the total distribution costs among its customers is hard to identify. The main issue in the work is the discussion about the methods developed for the supplier to calculate a cost-to-serve for each one. The simplest allocation methods that distribute the costs proportional to customer distance to the depot, storage capacity and consumption rate do not take into account the synergies between the customers and do not represent the true cost-to-serve. The obtained costs in a set of customers may indicate that an IRP solution is not the best if the internal relations between customers are considered. Additionally, the solution may lead to identify the low cost or high profit customers or show in which areas new clients would help to improve the company results.

In this chapter, several cost allocation mechanisms are proposed to be used in the context of the IRG. Özener (2008) classifies the cost allocation methods into three groups: Proportional methods, Per-route based methods and Duality based methods.

Proportional methods are generally easy to compute due to the fact that they do not take into account the effect of the synergies among the customers. In fact, they are the most commonly used methods in practice due to their simplicity. On the other hand, the methods in the other two groups consider the effect of customer synergies in some way; they provide more accurate cost-to-serve values, but the computation is though harder.

Another key aspect in the search of the solution is stability, as is a very restrictive condition. If a cost allocation is stable, the allocated cost to a subset of customers cannot be greater than the standalone cost of that subset. In fact, for most practical problems stable and budget balanced allocations do not exist. If the stability condition is not satisfied, the percentage deviation of the allocated cost of the subset from its standalone cost, the instability value of the subset, can be calculated.

- *Proportional Method*. This method provides a fast cost allocation taking into account some important factors such as customer distance to the depot; storage capacity and consumption rate. But it ignores other factors such as the synergies among the customers. The distance to the depot is the most relevant factor since the cost to be allocated is the total transportation cost of the problem. Therefore, the cost-to-serve value of a customer should be positively correlated to the customer's distance to the depot. Storage capacity and consumption rate are also significant. The allocated cost of a customer should be negatively correlated to the delivery period.
  The distances to other customers would reflect the geographical synergies between the customers. However, the estimation is difficult because in advance the customers actually grouped together on a route are not known; the synergies among customers will depend on the different storage capacities and consumption rates of the customers and the truck capacity for the route.
  The procedure, called PCAM (Proportional Cost Allocation Method), is very simple and efficient. Given an IRP instance, it provides in a very short time a balanced allocation even for very large instances of the problem. Briefly, this method allocates the total cost proportionally to the individual costs of the customers.
- *A Per-route-Based Cost Allocation Method*. This method is expected to perform better than the above proportional cost allocation method, since the cost allocation is based on the optimal routes that actually compose the total cost. It considers the synergies among the customers on a particular route, which will be called intra-route synergies. If the cost of each route is completely allocated among the served customers, the cost allocation will be balanced.
  The per-route-based cost allocations can be calculated quite efficiently, especially when the number of customers in the delivery routes is very limited. For instance, if every route has at most 4 customers, the general IRG is reduced to several small games with 4 customers. In this case, even if the number of routes is high, the computational effort required will be relatively low because a 4-player game is not computationally difficult to solve.
  Özener (2008) adapts to the IRG the "moat packing" cost allocation method proposed by Faigle et al. (1998) for the TSG. He argues that it can be easily modified to include the delivery volume information. Faigle et al. (1998) take into account the Held-Karp relaxation of the TSP to obtain cost allocations from moat packing using the duality in linear programming. It is considered a TSG with a cost matrix satisfying the triangular inequality.

The moat packing is introduced in the following way. The group of cities (also a city) is surrounded by nonintersecting moats in a TSP instance. If a vehicle has to reach the cities outside of a particular moat, it is obliged to cross twice the moat. The customers outside any given moat are equally responsible for the vehicle to cross over that moat. As the number of maximum moat packings gives a lower bound on the optimal cost of the TSP, each one of the moat packings can be used to allocate the cost among the cities visited.
A linear program is solved to determine the moat packing with maximum width. It prevents from the intersection of two moats. After the moat packing with maximum total width is known, the allocated cost is determined by distributing arbitrarily twice the width of any moat among the cities outside of the moat. For the IRG, the values for the customers may be different because they also reflect the influence on the execution of a route by the delivery volumes.
The ideal delivery volume to a customer is computed. This value is the minimum of the truck capacity and the storage capacity at the customer location. Therefore, if both customers would be at the same location, a customer that receives the ideal delivery volume should be allocated a higher cost than a customer that receives the residual amount in the vehicle. To allocate the cost, a ratio per customer of the actual volume delivered to a location by the route and the ideal delivery volume is computed.
The procedure is called MPCAM (Moat Packing Cost Allocation Method) and is composed of five steps. The first one identifies the set of optimal delivery routes. And for each route, the following steps are executed: the ratio is calculated, the dual is solved to obtain the optimal nested moat packing and the associated values using the procedure discussed above; the allocated cost of a customer along the route is computed… Finally, in the last step, once given the allocated cost of each customer for every delivery route, the final cost allocation of customers is obtained by adding up the cost allocations from individual routes. This method requires that the optimal cost and the optimal delivery routes of the IRP are known.
- *Duality Based Cost Allocation Methods*. Özener (2008) developed three different models that yield upper and lower bounds and an approximation to the optimal objective function value of the IRP. For several games based on combinatorial optimization problems, the relationship between its core, and the dual of the LP-relaxation of an Integer Program formulation of the problem is established. But instead of using just one LP-relaxation, he considers four different Linear Programs: the LP-relaxation of the mixed integer programming model (MIP-LP); the pattern selection LP model (PSLP); the LP-relaxation of the set partitioning model (SPM-LP); and the LP-relaxation of the set covering model (SCM-LP). Although none of the dual problems corresponding to these linear programs (DMIP-LP, DPSLP, DSPM-LP, DSCM-LP, respectively) will provide exactly the desired cost allocation, their solutions are modified to construct a cost allocation for the IRG. These duality-based methods are computationally efficient, since they only require that an LP is solved and possibly performing a scaling up operation to obtain a balanced cost allocation.

Generic cost allocation methods are not explored because any approach that computes the actual cost allocations obtained by the nucleolus or the Shapley value has an exponential number of coalitions and therefore is expected to have an exponential running time. For each coalition a NP-hard problem must be solved to compute exactly the nucleolus or the Shapley values.

12.5.2.2 Some Results

The experiments to evaluate the performance of the methods are based on random instances (Özener 2008). 50 different instances with 25 and 50 customers are generated, whose customers are located over a 1,000 * 1,000 square. The results are discussed in terms of computational efficiency on the stability of the generated allocations. Nevertheless, in Özener et al. (2013) three new instances with 26, 70, and 80 customers, respectively, are added. For these last instances, they prefer to analyze customers for which the cost allocations of the two methods differ significantly.

Going deep into the random instances, there are two types of customers with high or low storage capacities. Let Q the capacity of the vehicle and let S_i the finite storage capacity for the facility of customer i. The ratio Q/S_i for customer i falls within [0.5, 0.75] for high storage capacity customers or [1.25, 1.5] for those with low capacity. Similarly, the ratio of the consumption rate respect to the storage capacity is either within [0.5, 0.75] or [1.25, 1.5] if the customer belongs to the group with relatively low or high consumption rate. The planning horizon in all the instances is 100 periods and each period may include several subperiods.

Besides, there are clusters of customers. Specifically, in the instances with 25 customers, there are 3 clusters, and in the instances with 50 customers, there are 4 clusters. In both cases, 45 % of the customer locations fall within the clusters.

Song and Savelsbergh (2007) indicate that for distribution systems, such as in industrial gas distribution, it is common to have a few customers along a delivery route. For real-life size problems, this observation is used to limit the number of feasible patterns. Using this practical principle, Özener (2008) limits the number of stops along a route to at most 4 customers; then, automatically the number of feasible patterns is reduced.

The solution quality is evaluated with the maximum percentage instability of a cost allocation method. To calculate the exact percentage instability of a method, it is necessary to calculate the characteristic function values for all coalitions of the collaboration, which requires solving an IRP for each coalition. For this reason, he tests the stability of coalitions of size only up to 4 customers. Even with this simplification, this task requires evaluating 251,175 coalitions for the instances with 50 customers.

Different objective functions are used, but the best results are achieved with the approximation of the IRP optimal result by the set covering model (SCM-LP). The results of instability values for the different methods are shown in the Tables 12.3 and 12.4.

Table 12.3 Average, average of the maximum and maximum of the maximum percent instability of the methods (instances with 25 customers)

	PCAM	MPCAM	DPSLP	DSPM-LP	DSCM-LP
Average	5.86	3.96	3.90	0	0
Maximum	32.66	16.69	14.16	0	0
MAX (maximum)	51.26	36.44	21.93	0	0

Table 12.4 Average, average of the maximum and maximum of the maximum percent instability of the methods (instances with 50 customers)

	PCAM	MPCAM	DPSLP	DSPM-LP	DSCM-LP
Average	5.32	2.69	2.65	0	0
Maximum	32.91	13.52	11.82	0	0
MAX (maximum)	49.64	23.21	22.02	0	0

12.6 Conclusions

Cooperative game theory plays an important role in the redistribution of costs or profit in many real situations, such as those using VMI and related or similar policies. In this typically situations, a reduction in the total cost or an increase in the total profit can be achieved through cooperation. The case of IRP is a good example in this sense. VMI allows a significant reduction of the total distribution cost; however, the issue of allocating this benefit needs to be resolved. A "linear" allocation of these benefits to the players might not be fair and might lead to situations where some players become reluctant to cooperate. Game theory solves this sensitive issue, and so, it is admitted to the practice. The two cases discussed in the last section of this chapter provide a good motivation to such implementations where game theory is integrated with current operations management and supply chain optimization problems.

References

Adelman D (2004) A price-directed approach to stochastic inventory/routing. Oper Res 52(4):499–514

Aghezzaf EH, Raa BH, Van Landeghem H (2006) Modeling inventory routing problems in supply chains of high consumption products. Eur J Oper Res 169:1048–1063

Andersson H, Hoff A, Christiansen M, Hasle G, Lokketangen A (2010) Industrial aspects and literature survey: combined inventory management and routing. Comput Oper Res 37(9):1515–1536

Anupindi R, Bassok Y, Zemel E (2001) A general framework for the study of decentralized distribution systems. Manuf Ser Oper Mgmt 3(4):349–368

Archibald TW, Black D, Glazebrook KD (2009) An index heuristic for transshipment decisions in multi-location inventory systems based on a pairwise decomposition. Eur J Oper Res 192:69–78

Archibald TW, Black D, Glazebrook KD (2010) The use of simple calibrations of individual locations in making transshipment decisions in a multi-location inventory network. J Oper Res Soc 61(2):294–305

Arshinder S, Kanda A, Deshmukh SG (2008) Supply chain coordination: perspectives, empirical studies and research directions. Int J Prod Econ 115:316–335

Banerjee A, Burton J, Banerjee S (2003) A simulation study of lateral transshipments in a single supplier multiple buyers supply chain network. Int J Prod Econ 81–82:103–114

Borm P, Hamers H, Hendrickx R (2001) Operations research games: a survey. TOP 9(2):139–199

Bowersox D, Closs D, Cooper MB (2002) Supply chain logistics management. Mcgraw-Hill/Irwin Series Operations and Decision Sciences

Burton J, Banerjee A (2005) Cost parametric analysis of lateral transshipment policies in two-echelon supply chains. Int J Prod Econ 93–94:169–178

Campbell AM, Clarke LW, Kleywegt A, Savelsbergh MWP (1998) Inventory routing. In: Crainic TG, Laporte G (eds) Fleet management and logistics. Kluwer Academic Publishers, Dordrecht

Cao DB, Silver EA (2005) Dynamic allocation heuristic for centralized safety stock. Naval Res Logist 52(6):513–526

Chan C, Lee H (2005) Successful strategies in supply chain management. Hershey, IGI Global, doi:10.4018/978-1-59140-303-6

Christopher M (1998) Logistics and supply chain management: strategies for reducing cost and improving service. Financial Times-Prentice Hall, London

Dror M, Ball M (1987) Inventory-routing: reduction from an annual to a short period problem. Naval Res Logist Quart 34:891–905

Faigle U, Fekete SP, Hochstattler W, Kern W (1998) On approximately fair cost allocation in euclidean tsp games. Or Spektrum 20(1):29–37

Gaur V, Fisher M (2004) A periodic inventory routing problem at a supermarket chain. Oper Res 52(6):813–822

Ghiani G, Laporte G, Musmanno R (2004) Introduction to logistics systems planning and control. Wiley, Hoboken

Grahovac J, Chakravarty A (2001) Sharing and lateral transshipment of inventory in a supply chain with expensive low demand items. Manag Sci 47(4):579–594

Granot D, Sosic D (2003) A three stage model for a decentralized distribution system of retailers. Oper Res 51(5):771–784

Held M, Karp RM (1970) The traveling-salesman problem and minimum spanning trees. Oper Res 18(6):1138–1162

Held M, Karp RM (1971) The traveling-salesman problem and minimum spanning trees: part II. Math Program 1(1):6–25

Herer YT, Tzur M, Yücesan E (2006) The multi-location transshipment problem. IIE Trans 38:185–200

Hu X, Duenyas I, Kapuscinski R (2007) Existence of coordinating transshipment prices in a two-location inventory model. Manag Sci 53(8):1289–1302

Hu J, Watson W, Schneider H (2005) Approximate solutions for multi-location inventory systems with transshipments. Int J Prod Econ 97:31–43

Kleywegt AJ, Nori VS, Savelsbergh MWP (2002) The stochastic inventory routing problem with direct deliveries. Transp Sci 36(1):94–118

Kukreja A, Schmidt CP (2005) A model for lumpy demand parts in a multi-location inventory system with transshipments. Comput Oper Res 32(8):2059–2075

Lau RSM, Xie J, Zhao X (2008) Effects of inventory policy on supply chain performance: a simulation study of critical decision parameters. Comput Ind Eng 55:620–633

Mateo M, Aghezzaf EH, Vinyes P (2012) A combined inventory routing and game theory approach to solve a real-life distribution problem. Int J Bus Perform Supply Chain Model 4(1):75–89

Minner S, Silver EA (2005) Evaluation of two simple extreme transshipment strategies. Int J Prod Econ 93–94:1–11

Minner S, Silver EA, Robb DJ (2003) An improved heuristic for deciding on emergency transshipments. Eur J Oper Res 148:384–400

Özener OÖ (2008) Collaboration in transportation. Ph.D. thesis, Georgia Institute of Technology, Atlanta

Özener OÖ, Ergun Ö, Savelsbergh M (2013) Allocating cost of service to customers in inventory routing. Oper Res (in press)

Paterson C, Kiesmüller G, Teunter R, Glazebrook K (2011) Inventory models with lateral transshipments: a review. Eur J Oper Res 210(2):125–136

Raa B, Aghezzaf EH (2009) A practical solution approach for the cyclic inventory routing problem. Eur J Oper Res 192:429–441

Rudi N, Kapur S, Pyke DF (2001) A two-location inventory model with transshipment and local decision making. Manag Sci 47(12):1668–1680

Schmeidler D (1969) Nucleolus of a characteristic function game. SIAM J Appl Math 17(6):1163–1170

Shapley LS (1953) A value for n-person games. In: Tucker AW and Luce RD (eds) Contributions to the theory of games II. Princeton University Press, Princeton, pp 307–317

Shen ZJM (2007) Integrated supply chain design models: a survey and future research directions. J Ind Manag Optim 3(1):1–27

Simchi-Levi D, Kaminsky P, Simchi-Levi E (2000) Designing and managing the supply chain. McGraw-Hill, New York

Slikker M, Fransoo J, Wouters M (2005) Cooperation between multiple newsvendors with transshipments. Eur J Oper Res 167:370–380

Song JH, Savelsbergh MWP (2007) Performance measurement for inventory routing. Transp Sci 41(1):44–54

Southard PB, Swenset SR (2008) Evaluating vendor-managed inventory (VMI) in non-traditional environments using simulation. Int J Prod Econ 116:275–287

Stadtler H, Kilger C (2005) Supply chain management and advanced planning: concepts, models, software and case studies. Springer, Berlin

Tagaras G, Vlachos D (2002) Effectiveness of stock transshipment under various demand distributions and non-negligible transshipment times. Prod Oper Manag 11(2):183–198

Tempelmeier H (2011) Inventory-management in supply networks – Problems, models, solutions. 2nd (Edn), Norderstedt: Books on Demand. ISBN 9783842346772

Vinyes P (2007) Anàlisis de la cadena de subministre d'una cadena de franquícies basada en la teoria de jocs. Master Thesis ETSEIB, UPC

Waller M, Johnson ME, Davis T (1999) Vendor-managed inventory in the retail supply chain. J Bus Logist 20(1):183–203

Wee KE, Dada M (2005) Optimal policies for transshipping inventory in a retail network. Manag Sci 51(10):1519–1533

Wong H, Van Houtum GJ, Cattrysse D, Van Oudheusden D (2009) Multi-item spare parts systems with lateral transshipments and waiting time constraints. Eur J Oper Res 171:1071–1093

Wu YN, Cheng TC (2008) The impact of information sharing in a multiple-echelon supply chain. Int J Prod Econ 115:1–11

Zhao H, Deshpande V, Ryan JK (2006) Emergency transshipment in decentralized dealer networks: When to send and accept transshipment requests. Naval Res Logist 53:547–567

Zhang C, Zhang C (2007) Design and simulation of demand information sharing in a supply chain. Simul Model Pract Theor 15:32–46

Zhong YQ, Aghezzaf EH (2012) Modeling and solving the multi-period inventory routing problem with constant demand rates. In: 9th international conference of modeling, optimization and simulation (MOSIM'12), Bordeaux, France

Chapter 13
Winner Determination in Multi-unit Procurement Auctions with Volume Discount Bids and Lead Time Constraints

D. K. Verma, N. Hemachandra, Y. Narahari and J. D. Tew

Abstract In this chapter, we consider the problem of determining an optimal set of winning suppliers in a procurement auction where the buyer wishes to procure high volumes of a homogeneous item in a staggered way in accordance with a predefined schedule and the suppliers respond with bids that specify volume discounts and also delivery lead times. We show that the winner determination problem, which turns out to be a multi-objective optimization problem, cannot be satisfactorily solved by traditional methods of multi-objective optimization. We formulate the problem first as an integer program with constraints capturing lead time requirements and show that the integer program is an extended version of the multiple knapsack problems. We discover certain properties of this integer program and exploit the properties to simplify it to a 0–1 mixed integer program (MIP), which can be solved more efficiently. We next explore a more efficient approach to solving the problem using a linear relaxation of the 0–1 MIP in conjunction with a greedy heuristic. Using extensive numerical experimentation, we show the efficacy of the 0–1 MIP and the proposed heuristic.

Keywords Procurement auctions · Volume discount auctions (VDA) · Lead time constraints · Multi-attribute auctions (MAA) · Supply curve (SC) · Knapsack problem (KP) · Integer program (IP) · Mixed 0–1 integer program (0–1 MIP)

D. K. Verma · Y. Narahari (✉)
Department of Computer Science and Automation, Indian Institute of Science, Bangalore, India
e-mail: hari@csa.iisc.ernet.in

N. Hemachandra
Industrial Engineering and Operations Research, Indian Institute of Technology, Bombay, India

J. D. Tew
Cincinnati Innovation Lab, Tata Consulting Services, Milford, OH, USA

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_13,

13.1 Introduction

Procurement is an important activity in supply chains arising in large manufacturing firms, retail chains, etc. Auction-based mechanisms are extremely relevant in modern day electronic procurement systems since they enable a promising way of automating negotiations with suppliers and achieving the ideal goals of procurement efficiency and cost minimization (Chandrashekar et al. 2007). Firms that need to procure a high volume of a homogeneous item, would like to minimize the total cost of procuring the required number of units. In such procurement scenarios, the suppliers (also called vendors) would compete with one another by offering volume discounts. Economies of scale render such discounts possible. In such cases, procurement auctions with volume discount bids (Chandrashekar et al. 2007; Hohner et al. 2003) become relevant. In addition to cost of procurement and availability of volume discounts, there could be several other allocation constraints (see for example Sandholm 2007). One such important allocation constraint is the lead time to procure a batch of an item as required by a production schedule. Determining the winning suppliers in a procurement auction with volume discounts, taking into account lead time constraints offers many conceptual and computational challenges and also has practical relevance. This aspect has not been addressed in the literature and this chapter attempts to fill this research gap.

13.1.1 Motivating Example

There is a single buyer (or firm) who wants to purchase a certain number of homogeneous items from n suppliers (vendors), the set of suppliers being $N = \{1, 2,\ldots,n\}$. In any practical auction as discussed in Hohner et al (2003) and Eso et al. (2001) when the transaction volume is large the suppliers provide volume discounts in which the unit price decreases as we procure more number of units from that supplier. Such auctions are called volume discount auctions or supply curve auctions. In such an auction, the suppliers specify their bids as marginal decreasing piecewise linear price curves to capture volume discounts as shown in Fig. 13.1. In the bid depicted in Fig. 13.1, the supplier in question specifies that the minimum number of units he is willing to supply is 5 and for supplying up to 10 units, the per unit price charged is \$100. If the number to be supplied ranges from 11 to 20, then for the additional items beyond 10, the supplier would charge a discounted rate of \$80 per unit. If the number to be supplied ranges from 21 to 30, then the discounted rate is \$70 per unit for units from 21 to 30. Finally, the supplier charges accounted rate of \$40 units beyond 30, but he cannot supply beyond 45 units. Thus, if this supplier is required to supply 35 items, then the cost to the buyer would be $10 \times 100 + 10 \times 80 + 10 \times 70 + 5 \times 40$. If the volume discount bids (also called as supply curves in the literature), are of the above type, then there are a number of approaches suggested in the literature (see Sect. 13.2 for a survey of the relevant literature) for determining the set of winning suppliers and the numbers of units to be procured from the winning suppliers, so as to minimize the total cost of procurement.

In this chapter, we are interested in taking into account an important additional practical consideration that arises in real-world procurement scenarios. Typically, the buyer would like the units to arrive at the manufacturing or assembling facility according to some production schedule that is already worked out based on a number of practical and technical considerations. For example, the buyer may like to procure a total of 100 units by the first week, a cumulative total of 250 units by the second week, a cumulative total of 450 units by the third week, and so on. Table 13.1 shows a typical requirement from the buyer.

In response to this requirement from the buyer, the suppliers would be required to specify lead times in addition to volume discounts in their bids. In other words, a volume discount bid of the type shown in Fig. 13.1 will have to be enhanced with information about the lead times or delivery dates. Figure 13.2 shows such an enhanced bid. Here, the supplier does not supply less than 5 units. The unit price for supplying 5–10 units is 100 and these will be supplied earliest by the 5th day, and so on. We call such bids as Volume Discount Lead Time (VDLT) bids in this chapter. Accordingly, multi-unit procurement auctions with VDLT bids will be called VDLT auctions in this chapter. In principle and in practice, a supplier may submit several

Fig. 13.1 A volume discount bid

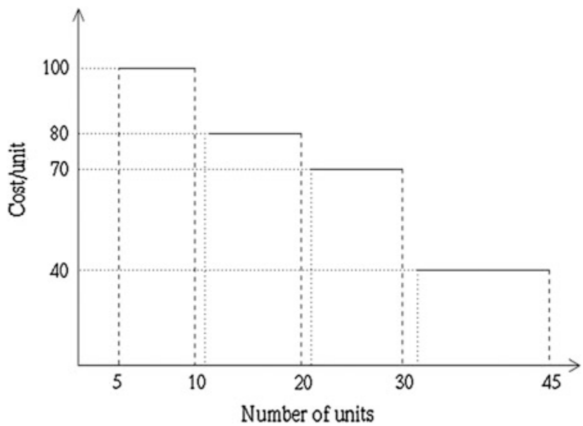
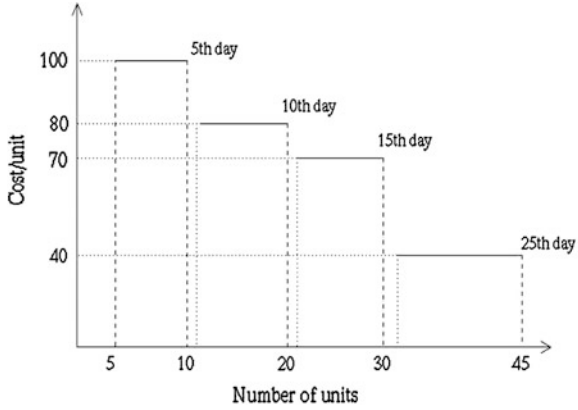


Table 13.1 Quantities required by the buyer

Interval (k)	Due date (D_k) (days)	Quantity (Q_k) required by the due date D_k
1	7	100
2	14	250
3	21	450
4	30	600
5	40	1,000

Fig. 13.2 A volume discount bid with lead time constraints



VDLT bids (trading off cost with lead time) in response to a buying requirement announced by the buyer. In this chapter, without loss of generality, we assume that each supplier submits exactly one VDLT bid. The case in which a supplier submits multiple VDLT bids could be immediately handled by replacing that supplier with as many virtual suppliers as the number of bids submitted by the supplier.

13.1.2 Contributions and Outline of the Chapter

The problem of optimal winner determination in procurement auctions with volume discount bids have been extensively studied in the literature. The next section will provide a review of the relevant literature there. However, to the best of our knowledge, procurement auctions with VDLT bids have been studied only sparsely (for example, see Kim et al. 2008). This chapter addresses this problem comprehensively and offers the following contributions:

- We bring out the limitations of traditional methods of multi-objective optimization in solving the winner determination problem of VDLT auctions.
- We model the winner determination in VDLT auctions as a multiple knapsack problem and formulate an integer program (IP). We derive an equivalent 0–1 mixed integer program (MIP), which is easier to solve, by exploiting certain nice properties of the IP.
- We propose a way to solve the above 0–1 MIP by first solving the LP relaxation of 0–1 MIP to get most of the solutions and then using a greedy heuristic to generate the remaining solution.
- We undertake extensive numerical experimentation to show the efficacy of the 0–1 MIP and also the heuristic.

The rest of the chapter is organized as follows. Section 13.2 is devoted to reviewing some preliminaries and all the relevant work. Section 13.3 defines the winner determination precisely, models it as a multiple knapsack problem, and derives an integer program. Certain properties of the solutions of this integer program are brought out, leading to an equivalent 0–1 MIP formulation. Section 13.4 describes an LP relaxation and a greedy heuristic to solve the 0–1 MIP. The numerical results are detailed in Sect. 13.5. Section 13.6 concludes the chapter and provides several directions for future work.

13.2 Background and Relevant Work

13.2.1 Multi-unit Auctions with Volume Discount Bids

In a procurement context where a single buyer and multiple sellers who wish to exploit scale economies are present, a volume discount auction is appropriate. Here suppliers provide bids as a function of the quantity that is being purchased (Davenport and Kalagnanam 2001; Hohner et al. 2003). The winner determination problem for this type of auction mechanism is to select a set of winning bids, where for each bid we select a price and quantity so that the total demand of the buyer is satisfied at minimum cost.

Kameshwaran and Narahari (2009) show that single item, single attribute, multi-unit procurement with volume discount bids leads to a piecewise linear knapsack problem. In this work, the authors have developed several algorithms (exact, heuristic-based, and fully polynomial time approximation schemes) for solving such knapsack problems. Goossens et al. (2007) present several exact algorithms for procurement problems with quantity discounts.

Kothari et al. (2003) consider single-item, single attribute, multi-unit procurement auctions where the bidders use marginal-decreasing, piecewise constant functions to bid for goods. The objective is to minimize the cost to the buyer. It is shown that the winner determination problem is a generalization of the classical 0/1 knapsack problem, and hence NP-hard. The authors also provide a fully polynomial time approximation scheme for the generalized knapsack problem. Dang and Jennings (2003) consider multi-unit auctions where the bids are piecewise linear curves. Algorithms are provided for solving the winner determination problem but these algorithms have exponential complexity in the number of bids.

Kumar and Iyengar (2006) consider a problem of optimal multi-unit procurement and characterize the optimal auction when the bids specify only the capacity (i.e., maximum number that can be supplied) and per unit cost. They also devise a one-shot get-your-bid procurement auction for the model they devise. Their work does not take into account volume discount bids. The recent work of Gautam et al. (2009) outlines an optimal auction in the presence of volume discount bids in a preliminary way.

None of the papers above treat the case of procurement auctions where, in addition to volume discounts, there are critical lead time constraints to be taken into account in determining the winning suppliers. The paper by Kim et al. (2008)

addresses this issue, albeit in a limited way. This chapter addresses this gap comprehensively by developing formulations and efficient algorithms for determining winners in a multi-unit procurement auction with volume discount bids and lead time constraints. Specifically, in this chapter we look at winner determination problem when there are lead time constraints in addition to volume discounts. We leave for future work the game theoretic issues when there are both volume discounts and lead time constraints.

13.2.2 Multi-attribute Auctions

Multi-attribute auctions relate to items that can be differentiated on several non-price attributes such as quality, delivery date, etc. Optimization problems for real applications often have to consider many objectives and thus we have to deal with a multi-objective (MO) optimization problem (Cohon 1978; Bichler and Kalagnanam 2001; Kameshwaran et al. 2006). If the objectives are conflicting, then the problem is to find the best possible design which still satisfies the opposing objectives. An optimal design problem must then be solved, with multiple objectives and constraints taken into consideration. Just as the concept of optimality plays an important role in the solution of single objective problems, the concept of non-inferiority or Pareto-optimality will serve a similar purpose for multi-objective problems. A feasible solution to a multi-objective programming problem is non-inferior if there exists no other feasible solution that will yield an improvement in one objective without causing degradation in at least one other objective. Some of the most popular non-inferior set generating techniques are the weighted method, the constraint method, the non-inferior set estimation method, and the Q-constraint technique.

In order to evaluate different offers for an item with different attribute levels we need to appeal to multi-attribute utility theory to provide a trade-off across these different attributes. A game theoretical model for multi-attribute auction is studied in (Bichler 2000) considering cost and quality. Multi-criteria procurement auction discussed in (Bellosta et al. 2004) requires the buyer to specify the aspiration point that expresses his desired values on the attributes and a minimum value. A technique for solving LP with multiple objectives using the model of zero-sum games with mixed strategies is presented in Belenson and Kapur (1973). Different approaches to solve multi-criteria programming is presented in Cohon (1978) like goal programming, constraint method, non-inferior set estimation method, and the multi-objective simplex method. A user interactive linear additive approach is presented in Zionts and Wallenius (1976), where, based on the feedback from user the weighting function of objective is changed. This type of approach is called interactive goal programming. An intuitive Q-constraint method is discussed in Haimes (1973), which converts the multi-objective to single-objective by considering other objectives to constraints. Integer linear programming with multiple objectives is discussed in Zionts (1977). To solve multiple objective integer program, Klamroth et al. (2004) use the Lagrangian duality theory.

13.2.3 Multi-objective Optimization

The process of simultaneously optimizing a collection of objective functions is an important focus within the field of engineering. The process is called multi-objective optimization. For a detailed review of multi-objective optimization, the reader is referred to Chap. 5 of Ravindran (2008). The general multi-objective optimization problem is formulated in Marler and Arora (2004) as follows:

Find $x = [x_1, x_2, \ldots, x_n]^T$

To minimize $F(x) = [F_1(x), F_2(x), \ldots, F_k(x)]^T$

Subject to $g_j(x) \leq 0 \quad j = 1, 2, \ldots, m$

where we try to find a n dimensional vector x so as to minimize objective $F(x)$ which is composed of k different objectives, subjected to m inequality constraints. The feasible design space X is defined as the set $X = \{x|g_j(x) \leq 0, j = 1, \ldots, m\}$. The feasible criteria space Z is defined as the set $\{F(x)|x \in X\}$. The decision space is defined in terms of decision variables whereas the criteria space is defined in terms of objective functions. Next, we discuss some of the multi-objective optimization technique like weighted method, constraint method, and ε-constraint method. Game theoretic (Belenson and Kapur 1973) and Lagrangian duality (Klamroth et al. 2004) have also been used for multi-objective optimization.

13.2.3.1 Weighted Method

Weighting the objectives to obtain non-inferior solutions is a traditional multi-objective solution technique. If someone were willing to articulate the value judgment of how much one objective is quantitatively important as compared to others, then the multi-objective problem could be reduced to a single-objective problem. The specification of w_i, which is called the weight on objective i, is equivalent to the identification of a desirable trade-off between different objectives, $i = 1, \ldots, k$. The above given multi-objective problem can now be formulated as:

Find $x = [x_1, x_2, \ldots, x_n]^T$

To minimize $F(x) = w_1 F_1(x) + \ldots + w_k F_k(x)$

Subject to $g_j(x) \leq 0 \quad j = 1, 2, \ldots, m$

Notice that the new objective is really valued and any two feasible points are comparable now. The solution would be the best compromise solution for the person who articulated the values of weights. For more details about the weighted method, the reader is referred to Chap. 5 of Ravindran (2008); Ravindran et al. (2010) and to the paper by Buyukozkan and Bilsel (2009).

13.2.3.2 Constraint Method

The constraint method (Cohon 1978) is perhaps the most intuitively appealing solution technique. It optimizes one objective while all of the others are constrained to some value. Suppose that instead of articulating a weight on lead time, the buyer stated a constraint, e.g., at least Q_1 number of units should be supplied in the first week, Q_2 in the second week, and so on. This type of constraint will restrict the feasible region that translates to a space above some horizontal line in the objective space. Proceeding in this manner, we can generate non-inferior solutions by solving a series of single-objective problems. The bounds that are placed on the constrained objectives, i.e., the Q_{i}'s, are considered as parameters to the solution process. Given a multi-objective problem with k objectives as above, the constrained problem is:

Find $x = [x_1, x_2, \ldots, x_n]^T$
Minimize $F_h(x)$
Subject to $g_j(x) \leq 0 \quad j = 1, 2, \ldots, m$

$$F_p(x) \leq L_p, \quad p = 1, 2, \ldots, h-1, h+1, \ldots, k$$

Here the hth objective is arbitrarily chosen for minimization. This formulation is a single-objective problem, so it can be solved by conventional methods, e.g., the simplex method in the case of linear programs. The optimal solution to this problem is a non-inferior solution to the original multi-objective problem if there exists any feasible solution for the multi-objective problem.

13.2.3.3 ε-Constraint Method

In this technique (Haimes 1973; Yokoyama et al. 1988) the original problem with k objectives is divided into k independent subproblems, each of which has respective objective function, $F_i(x)$. Let us denote by ε_i^0 the optimal value of ith subproblem. Now select one of the objective functions arbitrarily, say $F_1(x)$, out of the k functions, and designate this as objective function and the remaining $k-1$ objective functions will be treated as inequality constraints. Then, we can formulate the following optimization problem:

Minimize $F_1(x)$
Subject to $F_i(x) \leq \varepsilon_i, \quad \forall_i = 2, \ldots, k$
where $\varepsilon_i = \varepsilon_i^0 + \overline{\varepsilon_i}, \ \forall_i = 2, \ldots, k$

If the solution to above problem is feasible, then ε_i is decreased to tighten the constraint. This is repeated till the solution become infeasible and the minimum objective value is stored. Now the whole algorithm is iterated over the objective functions.

13.3 Volume Discount Auction with Lead Time Constraints

The inclusion of lead time in the volume discount auction gives the problem the form of multi-objective optimization problem. The buyer simultaneously wants to minimize the total cost of procurement as well as ensure that the time required for the supply of items satisfies the lead time requirement. The supply curves are specified by price and lead time schedules: decreasing step functions for each supplier. The step function corresponding to the supplier i is defined by a partition of function's domain into m_i intervals with corresponding unit price $p_i^j, \quad j = 1, 2, \ldots, m_i$ and lead time constraints $d_i^j = 1, 2, \ldots, m_i$. There are no gaps between the intervals and thus they can be represented by the set of $m_i + 1$ breakpoints as well. The quantity q_i^j represents the cumulative number of units the supplier i can deliver before d_i^j. The business constraints imposed by the buyer like lower and upper limit on total amount of units purchased from supplier i can be easily incorporated a priori before formulating the model by introducing these breakpoints into the appropriate supply curve and trimming away the curve left to the lower bound and discarding the breakpoints right to the upper bound. As already said, for the sake of simplicity we assume that the lead time specified in an interval corresponds to the whole of that quantity range. This assumption is not too strong since, if the supplier gives different lead times for single step we can consider it as two different steps with same price per unit item. The auction we consider can be described as follows.

- There is a set of suppliers $N = \{1, 2, \ldots, n\}$.
- The suppliers submit bids, $B = \{B_1, B_2, \ldots, B_n\}$. The bid quoted by supplier $i = 1, \ldots, n$ is a list, $B_i = \left[\left(q_i^1, p_i^1, d_i^1\right), \left(q_i^2, p_i^2, d_i^2\right), \ldots, \left(q_i^{m_i}, p_i^{m_i}, d_i^{m_i}\right)\right]$ where $q_i^0 < q_i^1 < q_i^2 < \ldots < q_i^{m_i}, p_i^1 > p_1^2 > \ldots > p_i^{m_i}$ and $d_i^1 \leq d_i^2 \leq \ldots \leq d_i^{m_i}$.
- Here the supplier i's valuation for the quantity range $\left(q_i^{j-1}, q_i^j\right)$ is p_i^j for each unit, and he can supply the lot earliest by d_i^jth time.

The problem of the buyer is to minimize the total cost of procurement as well as ensure that the required number of items is supplied within the lead times specified by the buyer.

First we wish to make the observation that the above auction is not equivalent a combinatorial auction problem (Kalagnanam and Parkes 2003); in fact, it turns out to be more general than a combinatorial auction. However, a very special case turns out to be the same as a combinatorial auction. We provide two examples to clarify these observations.

Example 13.1 Let us say the buyer wishes to procure 100 units by the 7th day and 250 units by the 14th day (which implies 150 units during the 8th to 14th days). Assume two suppliers who submit the following bids:

1. Supplier 1 promises to supply 50 units by the 7th day at a unit price of $ 10; 50 additional units by the 14th day with a unit price of $8.
2. Supplier 2 provides a bid which promises to supply 70 units by the 7th day at a per unit price of $9 and 100 additional units by the 14th day at a per unit price of $8.

Call the interval 0–7 days as item A and the interval 8–14 days as item B. Thus we need 100 units of A and 150 units of B. Supplier 1 is prepared to supply 50 units of A at price $10 each and 50 units of B at price $8 each. Supplier 2 is willing to supply 70 units of A at unit price $9 and 100 units of B at unit price $8. Thus the above problem becomes a multi-unit, multi-item auction for which there are efficient algorithms for winner determination.

Example 13.2 Let us say the buyer wishes to procure 100 units by the 7th day and 250 units by the 14th day (which implies 150 units during the 8th to 14th days). Assume two suppliers who submit the following bids:

1. Supplier 1 promises to supply 50 units by the 5th day at a unit price of $10; 50 additional units by the 10th day with a unit price of $8; and 50 additional units by the 14th day with a unit price of $7.
2. Supplier 2 provides a bid which promises to supply 70 units by the 7th day at a per unit price of $9; 50 additional units by the 10th day at a per unit price of $8; and 50 additional units by the 14th day at a per unit price of $6.

It is clear that a combinatorial auction cannot model the above situation because the demand intervals (specified by the buyer) and the supply intervals (specified by the suppliers) need not be the same. In fact, different suppliers may specify different supply intervals.

13.3.1 Limitations of Traditional Multi-objective Optimization Methods

In our problem of supply curve auction with lead time constraints, the two objectives are minimizing the cost of procurement and minimizing the lead time. We can state the objective of the auction mechanism using the weighted method given is Sect. 13.2.3.1 as follows:

$$\text{Minimize} \quad W_1[\{\text{Total\_Cost}\} + W_2\{\text{Total\_Lead\_Time}\}]$$

This is equivalent to:

$$\text{Minimize } [\{\text{Total\_Cost}\} + w\{\text{Total\_Lead\_Time}\}]$$

where $w = \frac{w_1}{w_2}$

Here we weigh the two objectives and combine it to give an overall objective. The weighting factor w_1 corresponds to the cost minimization objective while w_2

corresponds to the total lead time minimization objective. In this approach, it is quite easy to optimize the single objective problem. However, it is of serious doubt to assign weights in homogeneous manner when there exists trade-off among the objectives. Further, the meaning of the weighting factor is difficult to justify. The unit of w here is cost per day but specification of w can be quite difficult, as we are weighting the total lead time with total cost. This may be affected by the change is distribution of per unit cost and lead time specified by the suppliers.

There are many angles from which the lead time constraint can be viewed. In the most trivial situation, we want to minimize the final time of the arrival of the total number of units required by the buyer. This problem can be formulated using any of the techniques presented in Sect. 13.2.3 with the two objectives as cost minimization and minimization of maximum lead time. The second objective can be stated as a MiniMax formulation:

Minimize $[\{\text{Total\_Cost}\} + w\{\text{Max\_Lead\_Time}\}]$

The weighting factor w here gives the preference of unit lead time over cost. The advantage of this technique is that the specification of w is easy as it signifies the cost buyer has to pay for unit time delay in supply. But this method only minimizes the arrival of last unit, it does not take into account the arrival of other units, which may be of significant importance to the buyer in many situation. In the case of procurement by giant firms, the units are usually the raw inputs, and if the lead time is large the firm would not be able to start its production activity until the arrival of the last unit. These may lead to severe disadvantage to those who want to commence their activity with the arrival of units without waiting for all the units to arrive.

To take into account the above scenario, a more realistic situation can be thought as the buyer wants some rate of inflow of item (Bellosta et al. 2004), so that he can also simultaneously commence his activity while the other lots keep arriving. Let the buyer give quantity-deadline pairs as $\{(Q_k, D_k)\}, \quad \forall k = 1, 2, \ldots, t$. We cannot use weighting method when we have to meet the inflow requirement specified by the buyer. The other approach is to convert the lead time objective as constraint for each time interval, where the bound parameters that are placed corresponds to the number of units required by the buyer per time interval, leaving the cost minimizing objective as the single objective in the problem. A naive method to solve the optimization problem will be to solve the problem iteratively for each time interval, adding that time interval requirement only in the constraints, and reformulating the problem every time interval. Optimizing the cost in each time interval leads to an optimal solution for that time interval, but may eventually provide us a non-optimal solution for later intervals leading us to local optima and not the global one. The other way round we can formulate the problem as a single linear program having all the time interval constraints using constraint method of multi-objective optimization given in Sect. 13.2.3.2 as

Minimize $\{\text{Total\_Cost}\}$

Subject to $(\text{Quantity\_Procured\_}t\_k\} \geq Q_k, \quad \forall k = 1, 2, \ldots, t$

This approach works well only when the goal to be attained is well defined. This treatment, however, encounters a severe difficulty in the case where there exist trade-off relations among contradictory objectives. Since minimization of cost will lead to increase of lead time, the value attained by lead time if we minimize the cost is exactly the value specified by the buyer. Thus, the constraint method suffers from two major drawbacks:

- First, in case the buyer gives very strict deadlines, or the specified requirement cannot be met even by procuring items from all the suppliers, the solution may turn out to be infeasible.
- Second, if there are only a small number of suppliers with very high per unit cost, using this technique, we would be procuring the units at an unreasonably high cost to meet our requirement.

The ε-constraint method described in Sect. 13.2.3.3 too cannot be used for inflow requirement. The idea of traditional ε-constraint method is to iteratively increase the constraint bound by a predefined constant. The necessity to choose such a value represents also the main drawback of this approach. For a general problem the choice of this parameter is therefore not only difficult, it also influences the running time of the algorithm. As per (Laumanns et al. 2004) the running time of the original ε-constraint method is determined by the product of the ratio of the range to the minimum distance between two solutions in each objective. The ε-constraint technique yields a set of non-inferior solutions, any of which can be a candidate for the optimal solution. As a consequence, which solution should be taken as optimal can no longer be decided by just examining the objective functions themselves. One conceivable resolution for this ambiguity given in Yokoyama et al. (1988) may be to entrust a system operator with the final decision; he will choose the optimal solution out of the candidate solutions based on his personal experience and/or system operator policy. However, this way of decision can be quite subjective.

13.3.2 Mathematical Formulation

Having looked at various techniques for multi-objective optimization, and their drawbacks in our stated scenario we formulate the problem in a new and efficient way which will alleviate the problems of traditional techniques. Let the bid given by supplier i be as shown in Fig. 13.2. Here, we have

$$
\begin{aligned}
m_i = 4 \; q_i^0 &= 5 \\
\left(q_i^1, p_i^1, d_i^1\right) &= (10, 100, 5) \\
\left(q_i^2, p_i^2, d_i^2\right) &= (20, 80, 10) \\
\left(q_i^3, p_i^3, d_i^3\right) &= (30, 70, 15) \\
\left(q_i^4, p_i^4, d_i^4\right) &= (45, 40, 25)
\end{aligned}
$$

Here, the supplier i is not willing to supply any number of units less than 5. For the purchase of 5–10 units he will charge 100 per unit and can deliver the lot earliest by the 5th day. He will charge the next 10 units at a unit price of 80 and can deliver the next 10 units earliest by the 10th day, and so on. Note that this way of specifying the bids captures staggering of deliveries over time that may enable the buying firm to align these deliveries with their periodic production plans. We denote by m_i the number of steps or segments in the bid of supplier i.

As the buyer has specified the inflow rate required we can state the inflow as the constraints as in the constraint method, but along with the bound specified by the buyer we relax the bound by ε as is done in the ε-constraint method. Let ε_k be the number of units by which the requirement specified by the buyer for the kth time slot cannot be met. Here, the ε_k are variables in contrast to be parameters to be specified by the buyer in ε-constraint technique. Recall that Q_k is the number of units the buyer wants to procure by the end of the kth time interval, the actual quantity procured may be lower than Q_k by ε_k. This also helps to handle the situation which may be infeasible, i.e., if the requirement specified cannot be met even by procuring items from all the suppliers, then the solution does not turn out to be infeasible as in case of constraint method.

For each unit that he fails to meet the requirement, the buyer has to bear a loss of M which is added to the objective of the formulation. There is a substantial significance associated with the loss factor M. Even if there are enough units available from the suppliers to procure Q_k units of item in kth interval, the per unit cost of item may be higher than M and the buyer may not be willing to buy the unit at such high price; rather he would prefer to incur a shortage of procured units. This brings the concept of reserve price of buyer which is usually absent in all the traditional techniques. So, M signifies the preference relation of buyer between the two objectives of cost minimization and lead time minimization, and helps capture the trade-off between procurement cost and procurement lead time.

The loss factor M can be viewed as the weighting factor used in the weighted method of multi-objective optimization. While in the weighted method the weighting factor given by the buyer corresponds to the preference relation of the overall objectives, the M here represents the preference of lead time per unit of item over the cost per unit item, which gives a more intuitive preference relation that can be specified by the buyer easily as compared to the weighting factor of overall objectives. Thought here is an intimate relationship between the weighting and constraint methods, the specification of preference relation in our formulation is much easier for the buyer as compared to that of weighting method. Once the buyer has a clear view of his reserve price, he can specify the value of M as per his requirement. The notation used is summarized as follows.

$N = \{1, \ldots, n\}$	Set of n suppliers
m_i	Number of steps in the bid of seller
p_i^j	Unit price in interval j for seller i
q_i^0	Minimum number of units that the seller i is willing to supply
q_i^j	Cumulative number of units offered by the supplier by interval j
d_i^j	Lead time for interval j of seller i
x_{ij}	1 if interval j for seller i is selected, 0 otherwise
z_{ij}	Number of units to be procured in interval j from seller i

t	Number of demand intervals specified by the buyer
$k = 1, 2, \ldots, t$	Index for demand intervals
D_k	kth demand deadline specified by the buyer
Q_k	Cumulative number of units required by the buyer by the end of the interval D_k
ε_k	ε relaxation for number of units to be procured in the kth time interval
M	Loss incurred for shortage of each unit.

The IP formulation for the winner determination problem is given as follows:

- The x_{ij} decision variable indicates whether the amount of units purchased from supplier i falls into interval j or not. So, x_{ij} are of 0–1 type.
- Decision variables z_{ij} specify the amount of units purchased from partially sold interval j from the curve provided by supplier i. Thus, z_{ij} variables take non-negative integer values.
- The decision variable ε_k specifies the number of units by which we are short of the requirement of the buyer in the kth time interval.

Minimize

$$\sum_{i=1}^{n} \sum_{j=1}^{m_i} \left\{ z_{ij} p_i^j + x_{ij} c_i^j \right\} + M \sum_{k=1}^{t} \varepsilon_k$$

Subject to

$$z_{ij} - \left(q_i^j - q_i^{j-1} \right) x_{ij} \le 0 \quad \forall i \in N, \ \forall j = 1, 2, \ldots, m_i \tag{13.1}$$

$$\sum_{j=1}^{m_i} x_{ij} \le 1, \quad \forall i \in N \tag{13.2}$$

$$\sum_{i \in \varepsilon_k} \left\{ \sum_{j=z_{ik}+1}^{m_i} x_{ij} q_i^{z_{ik}} + \sum_{j=1}^{z_{ik}} \left(x_{ij} q_i^{j-1} + z_{ij} \right) \right\} \ge Q_k - \varepsilon_k \quad \forall k \in D \tag{13.3}$$

where

$$\begin{aligned}
c_i^j &= p_i^1 q_i^0 + \sum_{h=1}^{j-1} p_i^h \left(q_i^h - q_i^{h-1} \right), \quad \forall i \in N, \ \forall j = 2, \ldots, m_i \\
c_i^1 &= p_i^1 q_i^0 \quad \forall i \in N \\
s_{ik} &= \max\{ m : d_i^m \le D_k \} \quad \forall i \in N, \ \forall k = 1, 2, \ldots, t \\
s_k &= \{ i : s_{ik} > 0, i \in N \} \quad \forall k = 1, 2, \ldots, t \\
x_{ij} &= \{0, 1\} \quad \forall i \in N, \ \forall j = 1, 2, \ldots, m_i \\
\varepsilon_k &\in N \quad \forall k = 1, 2, \ldots, t
\end{aligned}$$

The first constraint (13.1) ensures that the value of z_{ij}, the quantity purchased from partially sold interval is positive and it is bound by the length of the interval itself. The second constraint (13.2) ensures that at most one of the intervals is selected as partially sold interval from each supplier. The third constraint (13.3) ensures that the requirement of the buyer is met in each time interval. So, we have $\sum_{i \in N} m_i$ constraints of type 1, N constraints of type 2, t constraints of type 3, $\sum_{i \in N} m_i$ 0−1 variables of form x_{ij}, $\sum_{i \in N} m_i$ integer variables of form z_{ij}, and t integer variables of the form ε_k.

13.3.3 Relaxing the Integer Program to a Mixed 0–1 Program

The mathematical formulation of the volume discount auction with lead time guarantees which we presented turns out to be an IP. In contrast to LP, which can be solved efficiently, IPs are in the worst case undecidable, and in many practical situations (those with bounded variables) are NP-hard. We present a few lemmas which will help to relax the IP to 0–1 MIP. A 0–1 MIP is a special case of an IP where the variables are required to be either 0 or 1 (rather than arbitrary integers) or linear.

The winner determination problem can be formulated as a generalized Knapsack Problem (KP) which is a generalization of the classical 0/1 KP. In the generalized KP, we have a knapsack size Q and a set of n lists, where the ith list is of the form,

$$A_i = \left[\left(\infty, q_i^0\right)\left(p_i^0, q_i^1\right), \left(p_i^2, q_i^2\right), \ldots, \left(p_i^{m_i}, q_i^{m_i}\right)\right]$$

where $p_i^1 > p_i^2 > \ldots > p_i^{m_i}$, $q_i^1 < q_i^2 < \ldots < q_i^{m_i}$ and p_i^j, q_i^j and Q are all integers.

Generalized Knapsack Problem: Determine a set of integers z_{ij} such that:

- for any i, at most one z_{ij} is non-zero,
- $z_{ij} \neq 0 \Rightarrow z_{ij} \in \left(q_i^{j-1}, q_i^j\right)$,
- $\sum_i \sum_j z_{ij} \geq Q$,
- $\sum_i \sum_j z_{ij} p_i^j$ is minimized.

One can very clearly see the connection between the generalized KP and the e-procurement problem. Here, each list corresponds to a bid and it represents multiple mutually exclusive quantity intervals. One can select at most one interval from each list and choose any quantity in that interval. Choosing a quantity in interval $\left(q_i^{j-1}, q_i^j\right)$ has a unique cost of p_i^j per unit. The goal is to procure at least Q units of item at minimum possible cost. Here, we observe an important property of an optimal knapsack solution.

Let us call each pair $b_i^j = \left(p_i^j, q_i^j\right)$ an anchor. The size of anchor is $b_i^j\ q_i^j$. In the feasible solution to generalized KP, we say an element, z_{ij} associated with b_i^jas critical if $z_{ij} = q_i^j$, else it is called midrange. A very important property is stated as a lemma in Dayama and Narahari (2007) is given below.

Lemma 13.1 *There exists an optimal solution to the generalized KP with at most one midrange element. All other elements are critical.*

Now using Lemma 13.1, we will present two more lemmas which will help us to show that the winner determination problem is actually an instance of 0–1 MIP.

Lemma 13.2 *The decision variables ε_k which corresponds to the number of units by which the requirement of the buyer cannot be met in time interval k, can only take integer values.*

Proof The ε_k can be non-zero if and only if:

1. the demand of the buyer cannot be met even by purchasing all the units possible from capable suppliers, or
2. the per unit cost given by the capable suppliers is more than the loss which the buyer has to bear for per unit shortage of item.

From Lemma 13.1 above, at a time only one anchor can be midrange, but since the requirement is more than what we had procured we can increase the units purchased from this midrange till it becomes critical. At this instant all z_{ij} variables will be zero. Now looking at the constraint (13.3) of the IP formulation, since all other terms are integers, ε_k can only take integer values. In the second case too, if per unit cost offered by the midrange supplier is less (greater) than the loss factor we can increase (decrease) the units purchased from that interval till it becomes critical. Thus, all anchors will be critical and the linear variables z_{ij} will be zero. Reasoning in a similar way, ε_k can take only integer values.

Lemma 13.3 *The decision variables z_{ij}, which correspond to the number of units purchased from interval j of curve supplied by supplier i, can only take integer values.*

Proof The RHS of the constraint (13.3) of the formulation, that is, the difference of Q_k and ε_k is an integer as both the operands are integers. In the LHS there will be only one non-zero z_{ij} as per Lemma 13.1, the remaining part of LHS consists of critical anchors which will take only integer values, so, the value of z_{ij} is also restricted to integers.

Using Lemmas 13.2 and 13.3, we are assured that the values attained by z_{ij} and ε_k are only integers. Thus we are left with x_{ij} decision variables which are 0–1 type, and the other decision variables z_{ij} and ε_k which are relaxed to take any linear value. Thus our problem of IP is reduced to 0–1 MIP problem which is computationally simpler than the IP.

13.4 Linear Relaxation and a Greedy Heuristic

In the previous section, we have relaxed the IP to a 0–1 MIP problem where the only variables x_{ij} which represent whether the quantity purchased from supplier i lies in interval j or not is of 0–1 type. While all other variables like z_{ij} which represent the actual number of units purchased from segment j of the SC given by supplier i, and ε_k which represent the number of unit by which we fall short to meet the requirement of buyer are linear. Yet, while this easier problem compared to IP, it is still a hard problem. In this section, we propose a computationally efficient way to solve the 0–1 MIP by first relaxing it to an LP and then using a heuristic to achieve a near optimal solution.

The success of the heuristic which we present next depends on some of the critical observations made with respect to the LP relaxation of 0–1 MIP.

- Winners with integer value of x_{ij} of LP relaxation of 0–1 MIP are also the winners of 0–1 MIP with the same selected segment.
- Quantities purchased from the winners with integer value of x_{ij} of LP relaxation of 0–1 MIP are same as that in case of 0–1 MIP.
- The winner corresponding to the midrange segments of 0–1 MIP are the only ones which are not present in the solution of LP relaxation of 0–1 MIP.
- Since for each demand interval there can be a maximum of one midrange segment as given by Lemma 13.1, the maximum number of winners present in the solution of 0–1 MIP which are not present in its LP relaxation is the number of demand intervals specified by the buyer.
- Since there are t demand intervals, and in each interval there can be at most one midrange segment, all the winners except at most t may be absent in the solution to the LP relaxation of 0–1 MIP.
- If the average number of units supplied by a typical supplier within demand interval k is q_k, then the maximum difference in total number of units obtained from integral solution of LP relaxation of 0–1 MIP and the original 0–1 MIP is $\sum_{k=1}^{t} q_k$.

Based on the above observations we can conclude that if the number of units to be procured is much greater than what can be supplied by a typical supplier, then most of the winners can be selected from the LP relaxation of the 0–1 MIP which are also present in the optimal solution of the 0–1 MIP.

Algorithm: SelectLPRWinner

The functioning of the algorithm can be summarized as follows: *SelectLPR-Winner* begins with the 0–1 MIP formulation of the bids and the demand, then it relaxes the 0–1 MIP to a LP and solves it using any standard LP Solver. From the solution set, all the non-integer solutions are rejected and all the integer solutions are added to a set of winners. Now after getting the winners, as above, we reduce the problem to select the remaining winners. We first remove the winners selected from the LP relaxation to get the reduced supply and also reduce the demand requirement of the buyer.

1. Formulate the 0–1 MIP from the bids B submitted by the sellers and the demand required $(Q_k, D_k) \quad \forall k = 1, 2, \ldots, t$.
2. Initialize the set Winners to the empty set, where each element is a tuple (Bidder, Quantity). Let Size be the number of Winners selected so far. Initialize Size $= 0$.
3. Relax the 0–1 MIP to LP by removing the integer constraint of x_{ij}, call it LPR.
4. Solve the LPR using any standard LP Solver.
5. For each segment with $x_{ij} = 1$, add i to Winners, i.e., set $\text{Bidder}_{\text{size}} = \{i | x_{ij} = 1\}$ and determine the number of units purchased from i.

$$\text{LPR\_}_{q_i} = \left\{ \left(q_i^{j-1} + z_{ij} \right) | x_{ij} = 1 \right\}$$

Add the tuple $((\text{Bidder}_{\text{size}}, \text{LPR\_}q_i)$ to Winners. Increment Size by 1.

6. Determine the number of units procured for each demand interval specified by the buyers. $\text{LPR\_Total\_}Q_k = \sum_{i=1}^{n} \text{LPR\_}q_{ik}$ where

$$LPR\_q_{ik} = \left\| \begin{Bmatrix} \{q_i^j | J = \max\{j : d_i^j \le D_k\}\} : \text{if LPR\_}q_i > q_i^j \\ \text{LPR\_}q_i : \text{if LPR\_}q_i \le {}_i^j \end{Bmatrix} \right\|.$$

7. From the original demand, determine the reduced demand set, by selecting only those intervals whose requirements are not yet met.

$$\text{Reduced\_}D = \{d_k | \text{LPR\_Totao\_}Q_k < Q_k\}$$

$$\text{Reduced\_}Q = \{(Q_k - \text{LPR\_Total\_}Q_k) | \text{LPR\_Total\_}Q_k < Q_k\}$$

Let the number of intervals in reduced demand in Reduced_D be t'.

8. Remove the bids submitted by the winner by making $\{q_i^j = 0 : \text{LPR\_}q_i > 0\}$ $\forall j = 0, 1, \ldots, m_i$. Call the new set of bids as B'.

SelectLPRWinner solves a LP which can be done in polynomial time. After solving the LPR it reduces the bids and demand which can also be done in some polynomial time that depends on the size of bids and number of demand intervals. Next we invoke the greedy heuristic *SelectGreedyWinners* to get the remaining winners.

13.4.1 A Greedy Heuristic

Now we present the heuristic *SelectGreedyWinners* which selects the winners with minimum cost. For each demand interval $(\text{Reduced\_}Q_k, \text{Reduced\_}D_k)$ it first selects a set of feasible segments from the bids. Feasible segments are the first segments from each bid which can meet the deadline for this demand interval as well the minimum number of sellable units for the bid $\left(q_i^0\right)$ is less than the number of units required and the price per unit $\left(p_i^1\right)$ in that segment is less than the loss factor M. A winner segment in a feasible set is the segment with minimum per unit price. If a winner segment is found with maximum number of units available in that segment more than the requirement of the demand interval, we truncate the bid by the requirement of that interval and move to the next interval. If a winner segment is found but the maximum number of units available is less than the requirement of the demand interval, it removes the selected winner segment from the corresponding bid and reduces the requirement of the demand interval with maximum number of units available in the winner segment and recreates the feasible set and searches for the next winner segment. This continues till either the requirement of the interval is met or the feasible set is empty. If the requirement is met, we move to the next demand interval and continue the selection of winner segments as above. If the feasible set is empty before the requirement of the interval is met, then we cannot meet the requirement of the interval and so we add the unsatisfied requirement to the next interval and move to next demand interval.

Heuristic: ***SelectGreedyWinners***

$$(\mathrm{B}', \text{Reduced\_}D, \text{Reduced\_}Q, \text{winners})$$

For each demand interval Reduced_D_k, $\quad \forall k = 1, 2, \ldots, t'$

1. Compute q_k where

$$q_k = \begin{cases} \text{Reduced\_}Q_k & : if \quad k = 1 \\ \text{Reduced\_}Q_k - \text{Reduced\_}Q_{k-1} & : if \; k = 2, \ldots, t'. \end{cases}$$

2. While $(q_k > 0)$

   2.1 Select winning seller I such that $q_i^0 \geq q_k$ and $d_I^1 < \text{Reduced\_}D_k$ and $p_I^1 = \min\{p_i^1\}, \forall i = N$.

   2.2 if winner selected with $p_I^1 \leq M$, add I to *Winner* set and reduce the requirement q_k

   (a) if $q_I^1 \geq q_k$, then $q_I^0 = 0 q_I^j = q_I^j - q_k \; \forall j = 1, 2, \ldots, m_i$ and $q_k = 0$ Add the tuple (I, q_k) to *Winners*, i.e., increment Size and k by 1. Go to Step 1.

   (b) else $q_I^0 = 0 q_I^j = q_I^{j+1}, p_I^j = p_I^{j+1}$ and $d_I^j = d_I^{j+1} \; \forall j = 1, 2, \ldots, m_i - 1$, $m_I = m_I - 1$ $q_k = q_k - q_I^1$ Add the tuple (I, q_I^1) to Winners, i.e., increment Size by 1. Go to Step 2.

2.3 If no Winner is selected Add the remaining quantity to next interval requirement.

$$\text{Reduced\_}Q_{k+1} = \begin{cases} \text{Reduced\_}Q_{k+1} + q_k & : if\ \ k = 1, \ldots, t' - 1 \\ \text{Reduced\_}Q_{k+1} & : if\ \ k = t' \end{cases}$$

Increment k by 1 and go to Step 1.

3. The set *Winner* contains all the winners.

For each demand interval we can meet the requirement in constant number of iteration of selecting the minimum cost winner from the feasible set, thus the complexity of *SelectGreedyWinners* will also be $O(t'N)$. So by relaxing the 0–1 MIP to an LP and solving the LP and the using the heuristic *SelectGreedyWinners* we have solved the VDLT problem in polynomial time. In the next section, we will compare the quality of the solution of 0–1 MIP and the above proposed solution.

13.5 Computational Results

In this section, we will present the results of computational experimentation carried out for VDLT auctions. We carried out the experiments by randomly generating the bids for sellers and the inflow requirement for the buyer. The LP and IP-formulated are solved using the commercial solver CPLEX from ILOG.

13.5.1 Performance of the 0–1 MIP Formulation

First we compare the performance of the 0–1 MIP formulation with that of the constraint method (CM). We have taken 100 bids each with specification stated as:

- $1 \leq m_i \leq 10, \ \forall i \in N;$
- $10 \leq q_i^0 \leq q_i^1 \leq \ldots \leq q_i^{m_i} < 100, \ \forall i \in N;$
- $10 \leq d_i^1 \leq d_i^2 \leq \ldots \leq d_i^{m_i} < 100, \ \forall i \in N;$
- $1000 \geq p_i^1 \geq p_i^2 \geq \ldots \geq p_i^{m_i} \geq 100, \ \forall i \in N.$

Varying Lead Time in Inflow Required

In Table 13.2 we give the comparison by changing the lead time specified by the buyer. The inflow requirement specified by the buyer can be given as

- $5 \leq t \leq 10$, where t is the number of intervals;
- $300 \leq Q_1 \leq Q_2 \leq \ldots \leq Q_t \leq 3000;$
- D_k Range specifies the range as specified in inflow required by the buyer;
- Q_t is the number of units required by the last interval;
- q_t is the number of units procured by the last interval;
- Procurement cost is the total cost of procurement by summing up the cost of each unit procured;

Table 13.2 Solutions obtained with 0–1 MIP formulation and constraint method for different lead times

D_k range	Q_t reqd.	Loss factor	Proposed 0–1 MIP			Constraint method		
			q_t proc.	Proc. cost	Proc. L.T	q_t procured	Proc. cost	Proc. L.T
2–20	2,895	300	229	162,159	2,455	Infeasible		
		600	743	576,249	9,477			
		900	1,024	788,677	14,289			
4–40	2,700	300	2,254	1,515,925	49,037	Infeasible		
		600	2,265	1,525,737	49,191			
		900	2,573	1,792,549	56,830			
6–60	2,663	300	2,088	1,206,404	49,037	2,663	1,627,101	56,830
		600	2,368	1,411,024	49,191			
		900	2,663	1,627,101	56,830			
8–80	2,746	300	1,750	1,033,366	40,851	2,746	1,539,068	84,011
		600	2,682	1,499,504	81,227			
		900	2,746	1,539,068	84,011			

- Procurement time is calculated by summing up the lead time of each unit procured as for procurement cost.

Table 13.2 shows that when the lead time requirement specified by the buyer is small, i.e., buyer wants the items early and sellers are not capable of supplying the units, the constraint method turns out to be inappropriate. The 0–1 MIP formulation proposed by us solves the problem by selecting the winners whoever is able to supply, even though may not able to meet the whole requirement. Similarly, in the case where there are enough sellers who can meet the requirement of buyer but their per unit cost is very high, the constraint method will procure the item without considering the unreasonably high price. But in our 0–1 MIP formulation, we can control this by varying the loss factor (M). As the loss factor is increased, the cost is traded-off with lead time. If we specify a high value of M, the solution is the same as what we achieve with the constraint method.

Varying Quantity Required in Inflow

In Table 13.3 we give the comparison with changing the quantity requirement specified by the buyer. The inflow requirement specified by the buyer can be specified as:

- $5 \leq t \leq 10$, where t is the number of intervals;
- $10 \leq D_1 \leq D_2 \leq \ldots \leq D_t \leq 1000$;
- Q_k Range specifies the range as specified in inflow required by the buyer;
- Q_t is the number of units required by the last interval;
- q_t is the number of units procured by the last interval;
- Procurement cost is the total cost of procurement by summing up the cost of each unit procured;

Table 13.3 Solutions obtained by the proposed 0–1 MIP formulation and constraint method for different demands

Q_k range	Q_t required	Loss factor	Proposed 0–1 MIP			Constraint method		
			qt Procurement	Procurement cost	Procurement L.T	qt	Procurement cost	Procurement L.T.
200–2,000	1,907	300	1,699	716,467	66,184	1,907	818,830	77,792
		600	1,907	818,830	77,792			
		900	1,907	818,830	77,792			
400–4,000	3,760	300	3,048	1,368,353	144,702	3760	1,795,571	177,055
		600	3,413	1,582,272	164,406			
		900	3,760	1,795,571	177,055			
600–6,000	5,662	300	5,189	1,386,179	181,204	Infeasible		
		600	5,662	3,749,702	193,481			
		900	5,662	3,880,266	188,418			
800–8,000	7,680	300	6,456	3,907,803	262,685	Infeasible		
		600	6,685	4,049,117	277,043			
		900	7,841	4,896,479	350,124			

- Procurement time is computed by summing up the lead time of each unit procured as for procurement cost.

Table 13.3 shows that when the quantity requirement specified by the buyer is small, that is, the buyer requires lower number of units than that can be jointly supplied by the sellers, we can achieve tradeoff between cost and lead time by varying the loss factor (M) in 0–1 MIP formulation. This avoids procurement at a very high cost which takes place in the constraint method. As the required quantities in each demand interval is increased, there may be a time interval when the requirement cannot be met even by procuring from all capable sellers. Under such circumstances, the constraint method fails to solve the problem. The 0–1 MIP formulation still solves the problem by procuring the maximum possible number units which have per unit cost less than M.

Trade-off Between Cost and Lead Time

The trade-off that can be achieved with 0–1 MIP formulation is presented in Fig. 13.3. For this, we considered a situation where the bids submitted by sellers have the following specification.

- $N = 100$;
- $1 \leq m_i \leq 10, \ \forall i \in N$;
- $10 \leq q_i^0 \leq q_i^1 \leq \ldots \leq q_i^{m_i} < 100, \forall i \in N$;
- $10 \leq d_i^1 \leq d_i^2 \leq \ldots \leq d_i^{m_i} < 100, \forall i \in N$;
- $2000 \geq p_i^1 \geq p_i^2 \geq \ldots \geq p_i^{m_i} \geq 100, \forall i \in N$.

The inflow requirement can be specified as:

$$[(Q_1, D_1)(Q_2, D_2)\ldots(Q_t, D_t)] = [(1350, 24)(1437, 25)(2176, 27)(2566, 66)(2648, 87)]$$

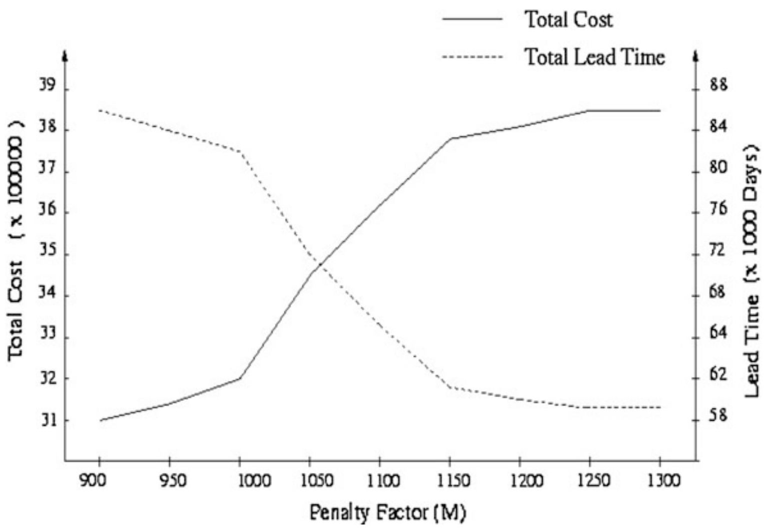


Fig. 13.3 Effect of changing loss factor on procurement cost and total lead time

The loss factor M signifies the buyer willingness to procure the item. A high value of M means that buyer is ready to procure the unit at unit price of M against its delay. M acts as a weight factor for delay of unit over its cost. As shown in the Fig. 13.3, as we increase M, the cost is traded off against the lead time, that is, the total procurement cost increases while the total time lead time decreases.

13.5.2 Performance of the Greedy Heuristic

After showing the advantages of 0–1 MIP against the constraint method, here we will compare the performance of the solution produced by the heuristic with that of optimal solution. The specifications of bids are same as given in Sect. 13.5.1. The demand curve can be specified as:

- $5 \leq t \leq 10$, where t is the number of demand intervals;
- $10 \leq D_1 \leq D_2 \leq \ldots \leq D_t \leq 100$;
- $300 \leq Q_1 \leq Q_2 \leq \ldots \leq Q_t \leq 3000$.

Figure 13.4 shows that the procurement cost of Heuristic, where first we solve the LPR of 0–1 MIP and then calls *SelectGreedyWinners*. The solution produced is of reasonably good quality, with an error of less than 5 %. Figure 13.5 shows that the time to solve the 0–1 MIP bears no correlation with the number of sellers, since solving an IP greatly depends on the structure of the constraints. If we use the heuristic (*SelectGreedyWinners*) after solving the LPR, we select the winners

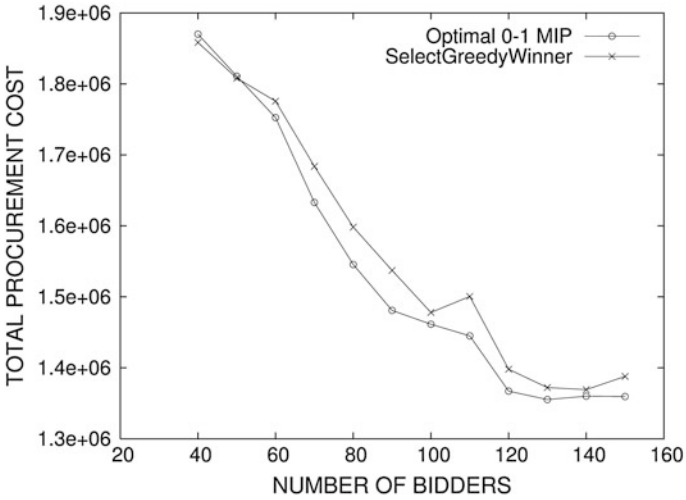


Fig. 13.4 Comparison of total cost of procurement using the greedy heuristic with that of 0–1 MIP formulation

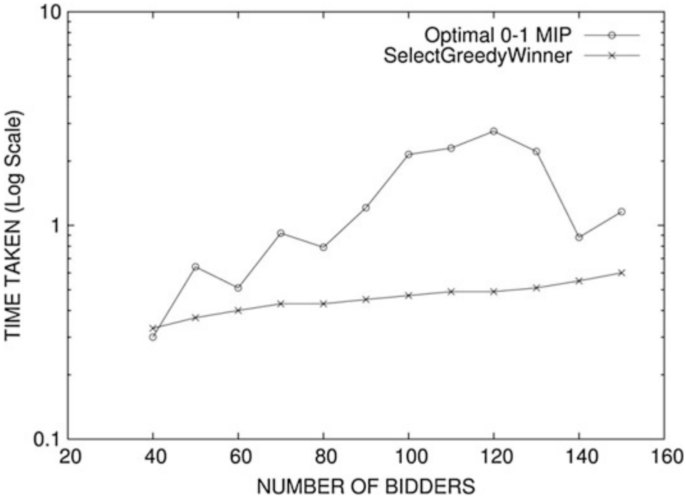


Fig. 13.5 Comparison of time taken to solve WDP with heuristic and optimal 0–1 MIP formulation

greedily in polynomial time $O(t'N)$.The complexity is proportional to the number of sellers, so as the number of sellers increases the time to solve also increases proportionally.

13.6 Summary and Future Work

In this chapter, we considered the problem of determining an optimal set of winning suppliers in a multi-unit procurement auction where the suppliers specify volume discounts and also delivery lead times. To the best of our knowledge, this is the first time both volume discounts and lead time constraints are considered together in a procurement auction setting in a detailed way. We showed that the winner determination problem, which turns out to be a multi-objective optimization problem, cannot be satisfactorily solved by traditional methods of multi-objective optimization. We formulated the problem first as an IP with constraints capturing lead time requirements and showed that the IP is an extended version of the multiple knapsack problem. We then simplified the IP to a 0–1 MIP, which is easier to solve. We also explored a more efficient approach to solve the problem using a linear relaxation of the 0–1 MIP, combining it with a greedy heuristic to obtain a high quality approximate solution in a fast way. Using extensive numerical experimentation, we showed the efficacy of the 0–1 MIP and the proposed heuristic.

We would like to suggest few directions in which this research can be taken further. The first direction is related to the approach for solving the winner determination problem, which happens to be a 0–1 MIP. One could look at more efficient ways of solving it by possibly exploiting the structure of the problem in some innovative ways. Another direction to pursue here would be to improve the heuristic. In fact, we tried a heuristic (Verma 2006), where we used the branch-and-bound method to search all possible solutions. The solution obtained with this heuristic was of higher quality with an error of about 1 %, but the time to solve was correspondingly higher. Moreover, the computational time was instance dependent as in the case of 0–1 MIP. Nevertheless, there is good scope for trying out other heuristics.

Second, the suggested winner determination problem in this chapter is restricted by an implicit assumption that there is a natural staggering in the requirement for the materials, which is a reasonable assumption. If the suppliers decide to deliver everything immediately in the first time interval itself, then the present formulation does not handle the situation that well. It would be worthwhile having warehouse capacity constraint to discourage early deliveries. This could be incorporated into the model.

The third direction to pursue would be look at the game theoretic angle. In this chapter, we have implicitly assumed that the buyer and the suppliers are honest and do not involve in strategic play while bidding. A more realistic setting would be to view them as rational players and use mechanism design theory to make the procurement auction robust to strategic play by the agents involved. There are many recent efforts in designing incentive compatible auctions in multi-unit procurement setting; we have already referred to these papers in the review of literature (Sect. 13.2.1). It would be interesting to pursue design of an incentive compatible auction in the current setting of volume discounts and lead time constraints. Pursuing the design of a cost minimizing auction subject to incentive compatibility and individual rationality would be an even more challenging direction.

References

Belenson S, Kapur K (1973) An algorithm for solving multi-criteria linear programming problems with examples. Oper Res Q 24(1):65–77

Bellosta M, Brigui I, Kornman S, Vanderpooten D (2004) A multi-criteria model for electronic auctions. In: Proceedings of the 2004 ACM symposium on applied computing, pp 759–765. ACM Press, New York

Bichler M (2000) An experimental analysis of multi-attribute auctions. Decis Support Syst 29(3):249–268

Bichler M, Kalagnanam J (2001) Winner determination in multi-attribute auctions. Technical report RC22478 (W0206-018), IBM

Buyukozkan G, Bilsel RU (2009) Multi-criteria models for e-health service evaluation. In: Dwivedi AN (ed) Handbook of research on information technology management and clinical data administration in healthcare, chapter 10. IGI Global, Hershey, pp 143–160

Chandrashekar TS, Narahari Y, Rosa CH, Kulkarni D, Dayama P, Tew JD (2007) Auction based mechanisms for electronic procurement. IEEE Trans Autom Sci Eng 4(3):297–321

Cohon (1978) Multi-objective programming and planning. In: Bellman R (ed) Mathematics in science and engineering, vol 140. Academic Press, New York, pp 115–127

Dang V, Jennings N (2003) Optimal clearing algorithms for multi-unit single-item and multiunit combinatorial auctions with demand-supply function bidding. Technical report, Department of Electronics and Computer Science, University of Southampton, UK

Davenport A, Kalagnanam J (2001) Price negotiations for direct procurement. Research report RC 22078, IBM Research, Yorktown Heights, NJ, USA

Dayama P, Narahari Y (2007) Design of multi-unit electronic exchanges through decomposition. IEEE Trans Autom Sci Eng 4(1):67–74

Eso M, Ghosh S, Kalagnanam J, Ladanyi L (2001) Bid evaluation in procurement auctions with piece-wise linear supply curves. Technical report IBM research report RC 22219, IBM

Gautam RK, Hemachandra H, Narahari Y, Prakash H, Kulkarni D, Tew JD (2009) An optimal mechanism for multi-unit procurement with volume discount bids. Int J Oper Res 6(1):70–91

Goossens DR, Maas JAT, Spieksma FCR, van de Klundert JJ (2007) Exact algorithms for procurement problems under a total quantity discount structure. Eur J Oper Res 178(2):603–626

Haimes Y (1973) Integrated system identification and optimization. In: Leondes C (ed) Control and dynamic system: advances in theory and application, vol 9. Academic Press, New York, pp 435–518

Hohner G, Rich J, Ng E, Reid G, Davenport A, Kalagnanam J, Lee H, An A (2003) Combinatorial and quantity-discount procurement auctions benefit, mars incorporated and its suppliers. Interfaces 33(1):23–35

Kalagnanam J, Parkes D (2003) Auctions, bidding, and exchange design. In: Simchi-Levi D, Wu SD, Shen ZM (eds) Supply chain analysis in the eBusiness area. Kluwer Academic Publishers, Norwell

Kameshwaran S, Narahari Y (2009) Efficient algorithms for non–convex linear knapsack problems. Eur J Oper Res 192(1):56–68

Kameshwaran S, Narahari Y, Rosa CH, Kulkarni DM, Tew JD (2006) Multi-attribute electronic procurement using goal programming. Eur J Oper Res 179(2):518–536

Kim T, Bilsel RU, Kumara S (2008) Supplier selection in dynamic competitive environments. Int J Serv Oper Inf 3(3–4):283–293

Klamroth K, Tind J, Zust S (2004) Integer programming duality in multiple objective programming. J Glob Optim 29(1):1–18

Kothari A, Parke DC, Suri S (2003) Approximately–strategy proof and tractable multi-unit auctions. In EC'03: proceedings of the 4th ACM conference on electronic commerce. ACM Press, New York, NY, USA, pp 166–175

Kumar A, Iyengar G (2006) Optimal procurement auctions for divisible goods with capacitated suppliers. Technical report TR-2006-01, Columbia University

Laumanns M, Thiele L, Zitzler E (2004) An efficient adaptive parameter variation scheme for matheuristics based on the epsilon-constraint method. Eur J Oper Res 169(3):932–942

Marler R, Arora J (2004) Survey of multi-objective optimization methods in engineering. Struct Multi-discip Optim 26(6):369–395

Ravindran A (2008) Operations research and management science handbook. CRC Press, Boca Raton

Ravindran A, Bilsel RU, Wadhwa V, Yang T (2010) Risk adjusted multi-criteria supplier selection models with applications. Int J Prod Res 48(2):405–424

Sandholm T (2007) Expressive commerce and its application to sourcing: how we conducted $35 billions of generalized combinatorial auctions. AI Mag 28(3):0

Verma DK (2006) Optimization models for multi-attribute auctions and exchanges. Technical report, Master of engineering dissertation, department of computer science and automation, Indian Institute of Science, Bangalore

Yokoyama R, Bae S, Morita T, Sasaki H (1988) Multi-objective optimal generation dispatch based on probability security criteria. IEEE Trans Power Syst 3(1):317–324

Zionts S (1977) Integer linear programming with multiple objectives. Ann Discret Math 1:551–562

Zionts S, Wallenius J (1976) An interactive programming method for solving multiple criteria problems. Manag Sci 22(6):652–663

Chapter 14
A Piecewise Linear Supply Chain Game for Manufacturing Network Formation

S. Mahjoub and J. C. Hennet

Abstract This chapter analyzes the process of forming a coalition within a corporate network. The objective of the partner companies is to create a multistage manufacturing system, which generates a chain of increased value from raw materials to end-user market. This process is studied by cooperative game theory, through the key problems of maximizing the total profit and distributing it among the members of the coalition. To construct a pay-off policy that is both stable and fair, the study proposes to represent the productive resources of the firms not only by their capacity, but also by the work in progress (WIP) generated by product flows. The proposed profit sharing rule is then constructed from the dual of the profit maximization problem. It is both efficient and rational, with more fairness than the Owen set policy of classical linear production games.

Keywords Corporate networks · Game theory · Manufacturing systems · Resource utilization · Clearing functions · Linear production games · Duality

14.1 Introduction

The concept of a manufacturing network is close to the concepts of virtual enterprise and extended enterprise. Powell and Grodal (2005) define an enterprise network as a group of companies maintaining formal relations between them in the form of contracts that can be rather informal or can manifest just an exchange of

---

S. Mahjoub
LEMNA, Nantes University, Chemin de la Censive du Tertre-BP 52231 44322 NANTES Cedex 3, France

J. C. Hennet (✉)
LSIS-CNRS, Saint–Jerome Campus, Aix-Marseille University, Avenue Escadrille Normandie Niemen 13397 Marseille Cedex 20, France
e-mail: jean-claude.hennet@lsis.org

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_14,

information between the different parties. Butera (1991) defines a similar concept as a set of firms linked to each other by a production cycle. The link is neither legal nor structural; it often takes the form of simple agreements. What these companies have in common is a powerful drive to functional cooperation. In recent years, companies have realized that forming networks could help them to meet the requirements of innovation and responsiveness required by a continuously changing environment. Pooling resources and sharing services create interdependencies. According to Paché and Bacus-Montfort (2003), a corporate network has a reticular form with no permanent leader. Its purpose is to combine the resources of multiple stakeholders to achieve a common goal, and even disappear as soon as the goal is reached. A manufacturing network appears as an organizational form that combines flexibility, competitiveness, and responsiveness. It therefore constitutes a new paradigm that may emerge as the dominant mode of organization. The effectiveness of this organizational form has been demonstrated in various sectors, including automotive, electrical, electronics, IT, construction, banking and insurance, aerospace, and food processing. There are many examples of alliances in today's economy, including in particular BMW, DaimlerChrysler, and General Motors for hybrid propulsion systems and the Renault and Nissan alliance.

The issue of forming a manufacturing network in a set of firms connected by information and logistic links can be modeled by the theory of cooperative games. As noted by Williamson (1981), most companies do not have sufficient internal resources to ensure economies of scale and reduce the uncertainty associated with entry into new markets. For this reason, the association with other companies is often the best solution to deal with these issues. Indeed, manufacturing networks allow for a better reactivity of production and a better mobilization of resources to cope with market fluctuations. The works of Cachon and Netessine (2004) and Nagarajan and Sošić (2008) provide convincing interpretations of manufacturing network design problems as cooperative games. Each applicant company is seen as a player seeking to maximize its utility function, which is its expected profit. The means to achieve this goal is the formation of a coalition of players, which involves the sharing of production resources to maximize the expected total profit obtained by the production and sale of finished products.

In a cooperative game, a coalition is a subset of the set of players. It is associated with an income that can be shared among its members. A cooperative game raises two basic problems:

- The problem of optimizing the income and determining the coalitions for which the optimum is reached;
- The problem of distributing the income between the players who belong to the optimal coalition.

In this context, methods of distribution of income are numerous. These methods include the method of equal sharing, which consists of the remuneration of all members of the coalition in an identical manner. Despite its simplicity, this method is not as fair as it seems to be, since it does not take into account the different resource contributions of the participants. At the other extreme, Gerchak and Gupta

(1991) have proposed a method of sharing the total income based on the contribution of each partner to the maximal total profit. Although this method allows a fair distribution, it does not generally ensure the stability and sustainability of the optimal coalition. The viability and performance of a coalition highly depend on the quality of the income sharing policy. Among the properties that a policy should possess to be fully accepted by the partners, we can mention efficiency, rationality, and equity. Cooperative game theory offers solution concepts related to these properties. The main concepts commonly used in the cooperative game context are the core, the Shapley value, and the nucleolus (see e.g., Osborne and Rubinstein 1994). This study focuses on sharing policies that belong to the core of the game. Under such a policy, the choice of the winning coalition is both efficient and rational for all the players.

Many examples of core allocation policies can be found in the literature on collaborative enterprise networks. Hartman et al. (2000) analyzed a supply chain consisting of several retailers facing random demands. The proposed model follows the single product newsvendor model. The authors supposed that retailers can pool their stocks and centralize their order quantities. By imposing certain conditions on the distribution function of demand, the authors showed that the core of the stocks centralization game is not empty. Then Muller et al. (2002) relaxed these restrictions and showed that the core of the stocks centralization game is never empty, independently of the demand distribution. In addition, they gave a necessary and sufficient condition for the core to be a singleton.

Hartman and Dror (2003) studied a cooperative game with several retailers facing stochastic and dependent demands. In particular, they assumed that the demands of different retailers are correlated and follow a normal distribution. Their study shows that the game is subadditive with a nonempty core if storage costs and stockout costs are identical for all the subsets of retailers. This result remains valid when demands are independent. In contrast, the core can be empty if storage and shortage costs are mutually different (Hartman and Dror 2005). Similarly, Meca et al. (2004) studied a supply chain formed by several retailers. Each retailer uses the economic order quantity policy for supply and subsequently meets the demand assumed deterministic. In addition, each retailer undergoes a linear storage cost and a fixed cost of supply. The authors noted that the total supply costs can be reduced to a single cost borne by the grand coalition of retailers. They showed that therefore the core of the retailers' cooperative game is not empty. In addition, they proved that the solution with proportional distribution of the total profit is in the core. In a similar framework of cooperation between retailers with centralized stocks, Slikker et al. (2005) have integrated transportation costs in the model and show that the core of the cooperative game is not empty.

In order to form a supply chain, enterprises should select the quantities of end-products that they plan to sell on the market and use their means of production in agreement with the product structure and the operation sequence of the goods to be manufactured. Such an arrangement should be performed in the most profitable manner for all the enterprises involved. The choice of the most efficient coalition of enterprises pooling their logistics and manufacturing resources is the main issue of the Linear Supply Chain Game (LSCG) studied in Hennet and Mahjoub (2010). The LSCG is an extension of the linear production game (LPG) studied by Shapley and Shubik (1972). A remarkable property of the LPG is the existence of a set of purely competitive profit sharing policies, known as the Owen set, which lies at the core of the game (Owen 1975). These policies are constructed from a dual program in which the optimal dual variables can be interpreted as the marginal costs of resources. One limitation of the standard LPG formulation is the difficulty to combine rationality and fairness in the allocation policy, due to the fact that marginal costs of resources drop to zero when their capacity is in excess in the coalition. There are several definitions of fairness in cooperative games. In the context of production games, the rewards of players can be considered fair if they are in the same proportion of their partial contribution in any coalition. This definition can be formally stated as the “balanced contributions property” defined in Myerson (1977). Then, according to this definition, the Shapley value allocation is the only fair allocation policy. The possibility for the Shapley value allocation to also be rational is closely related to convexity properties of the cooperative game. Unfortunately, linear production games are not convex in general (Hennet and Mahjoub 2011) and there are many such games in which the Shapley value allocation is not in the core of the game.

A milder definition of fairness has been proposed in Hennet and Mahjoub (2010). It requires that all the players who belong to the winning coalition should receive a strictly positive payoff. A sufficient condition for this property to hold true has then been obtained for the LSCG. One of the objectives of this chapter is to satisfy the same milder fairness condition, but in a more systematic manner, through a more precise and operational model of the saturation constraints on resources in manufacturing processes. This study considers manufacturing resources not only from the capacity viewpoint but also from their utilization conditions, by taking into account the influence of the workload on the throughput and introducing the cost of the work in progress (WIP) in the objective function. The value function of the chain integrates as a positive term the anticipated revenue to be obtained from the sale of the end-products on the market and as negative terms manufacturing costs and holding costs of all the products and components.

Then, the piecewise linear supply chain game (PLSCG) defined and studied in this chapter represents the saturation constraints on resources in a detailed manner, while preserving linearity of the model. The main reason for maintaining the linearity property is to remain in the scope of the LPG, with the resulting possibility of using the Owen set profit allocation rule as a rational profit allocation policy. The properties of the PLSCG demonstrated in the chapter allow for an easy computation by Linear Programming of the maximal profit achievable from the formation of a supply chain in the network of enterprises. Then, a stable profit allocation rule is constructed as a generalization to the PLSCG of the Owen set concept used for LPG. A practical advantage of the proposed PLSCG allocation policy over the classical Owen policy for the LSCG is that it improves the fairness of the profit allocation rule by integrating the costs of resource utilization before reaching the capacity limits.

Section 14.2 introduces the supply chain design model obtained from the product structure of the end-products to be sold on the market. Section 14.3 solves the PLSCG by optimizing the expected profit of the supply chain. A stable profit allocation policy is then constructed in Sect. 14.4. An illustrative numerical example is provided in Sect. 14.5, and Sect. 14.6 concludes the chapter.

14.2 A Supply Network Formation Process

14.2.1 Multistage Manufacturing Models

The concept of supply network concentrates on some major features of business organization in today's society. It characterizes a network of autonomous production units connected through an information and communication network and through a logistic network. In the recent literature, such a network is sometimes called "cloud of collaborative enterprises" or "cloud supply chain" (Lindner et al. 2010). Typically, the information network carries commercial proposals, products orders, manufacturing, and delivery protocols. The logistic network used to transport goods and products may be owned by enterprises that belong to the network or by subcontractors. In any case, a supply chain can be viewed as a multistage production and transportation system in which the different transformation stages are performed by different enterprises. Requirements planning models (Baker 1993) can then be used to define and distribute responsibilities and manufacturing orders among the partners. In this view, the product structure supports the enterprise network organization, especially under an extended view of the BOM (Bill of Materials), such as the G-BOM (Generic BOM), (Lamothe et al. 2005), integrating product families rather than simple products.

Consider a set of end-products (or families of end-products) $i \in \{1, \ldots, g\}$ and a perfectly competitive market. From the consumer's viewpoint, these products may be substitutable or not, and any quantity of them can be sold on the market at the fixed market prices: p_i for $i \in \{1, \ldots, g\}$. Consider now the set of all primary, intermediate, and end products: $i \in \{1, \ldots, g, \ldots, n\}$. Each production stage is supposed to have several input products but only one output product. The BOM technical matrix Π, is defined as follows: according to a given manufacturing recipe, production of one unit of product i requires the combination of components $l \in \{1, \ldots, n\}$ in quantities π_{li}.

In terms of modeling, a convenient graphical tool to represent multi-product multi-stage bills of materials is the gozinto graph proposed by Vazsonyi (1955). This graph defines a partial order in the set of products and a total order by classes of products that are called "levels." By definition, end products have level 0 and the level of product i, for $i = g + 1, \ldots, n$ is the maximal number of stages to transform product i into an end product. Accordingly, intermediate and primary products are numbered in the increasing order of their level. Under a level-consistent ordering of products, matrix Π has a lower triangular structure (Hennet 2003).

Additionally, multistage production by several producers highly differs from multistage production by a single producer because of the need for negotiation, contracts, and higher coordination requirements. It also carries new possibilities in the design stage for selecting partners, sharing resources, risks, and rewards.

Let $\mathcal{N}$ be the set of N enterprises who candidate to be a part of the supply chain to be created. Each candidate enterprise is characterized by its production resources: manufacturing plants, machines, work teams, robots, pallets, and storage areas. The supply chain model is formulated over a reference time horizon and in stationary conditions. Thus, lead times are not included in the model and material supply is supposed perfectly coordinated with manufacturing processes. Then, let $X = \left(\left(x_{ij}\right)\right)$ be the matrix of the quantities of product i produced (or obtained by exchange) at firm j and $y = (y_1, \ldots, y_n)^T$ be the output vector during the reference period. The components of this matrix and vector are the variables of the design problem. For simplicity, quantities per period (or throughputs) are supposed continuous: $X \in \Re_+^{n \times N}$, $y \in \Re_+^n$. The total product throughput vector, denoted $\omega = [\,\omega_1 \quad \cdots \quad \omega_n\,]^T$ is related to matrix X through summation relations (14.1):

$$\omega_i = \sum_{j=1}^{N} x_{ij}\, for \quad i = 1, \ldots, n. \tag{14.1}$$

Equations (14.1) for $i = 1,\ldots, n$ are summarized in vector form:
$\omega = X\mathbf{1}_N, \mathbf{1}_N$ being the unit vector of dimension N.

The output vector can be computed from the throughput vector by the following relation:

$$y = (I - \Pi)\omega, \text{ with } I \text{ the } n \times n \text{ identity matrix.} \tag{14.2}$$

From the structure of matrix Π, matrix $(I - \Pi)$ is regular and matrix $(I - \Pi)^{-1}$ is lower triangular (with 1 s on the diagonal) and nonnegative. Then, for a nonnegative output vector y, the global throughput vector is also nonnegative since it is expressed as follows:

$$\omega = (I - \Pi)^{-1} y. \tag{14.3}$$

14.2.2 Resource Capacity and WIP

Consider the R types of resources available in the network ($r = 1,\ldots, R$). The amount of resource r available for enterprise j is denoted as k_{rj}, and the resource capacity matrix is defined as: $K = \left(\left(k_{rj}\right)\right) \in \Re^{R\times N}$. A subset S of enterprises, with $S \subseteq \mathcal{N}$, can be represented by its characteristic vector $e_S \in \{0,1\}^N$ such that:

$$\left\langle \begin{array}{ll} (e_S)_j = 1 & \text{if } j \in S \\ (e_S)_j = 0 & \text{if } j \notin S \end{array} \right. \tag{14.4}$$

Let m_{ri} be the amount of resource r necessary to produce 1 unit of product i. $M = ((m_{ri})) \in \Re^{R\times n}$.

The first issue addressed in this chapter is how to represent resource capacity in a consistent manner with the resource saturation phenomena observed in practice. Basically, a resource such as a machine or transportation equipment is characterized by its decreasing efficiency relative to the load.

Classical capacity constraints used in aggregate production planning problems are simple saturation functions of the "all or nothing" type. Such capacity constraints are valid and will be used in our model. In particular, the capacity constraints restricted to a coalition $S \subseteq \mathcal{N}$ are written as:

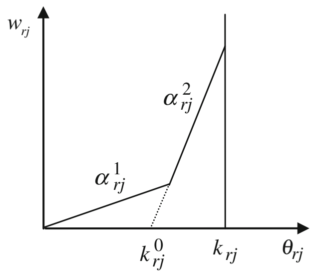
$$M(I - \Pi)^{-1} y \le K e_S \tag{14.5}$$

But constraints (14.5) are not sufficient to describe the saturation phenomena, because they do not represent the average workload in the system. Using the "Little law" (Little 1961), systems with saturation functions completely described by (14.5) are associated with constant lead times. However, as stressed in Karmarkar (1993), data show a super linear increase of lead times with capacity utilization, and this property is true at various magnitude levels and for any type of physical resources. In queuing theory, a lead time is classically decomposed into a processing time of constant expected value and a waiting time whose mean value increases with the mean population (or WIP, Work In Progress). Such convex lead time variations with the WIP are associated with concave output functions called clearing functions. This phenomenon has been largely ignored in the production planning literature (Asmundsson et al. 2003), probably because of the difficulty to integrate nonlinear constraints in planning problems that are already complex, with large number of variables. To keep our supply chain model linear, this nonlinearity will be represented by a piecewise linear function that represents the inverse clearing function (Fig. 14.1). This function expresses the WIP as a function of the actual throughput.

For resource r located at enterprise j, the WIP is denoted w_{rj}. The resource throughput, denoted θ_{rj}, is given by:

$$\theta_{rj} = \sum_{i=1}^{n} m_{ri} x_{ij} \tag{14.6}$$

Fig. 14.1 A piecewise linear inverse clearing function



The clearing function of Fig. 14.1 is represented by the following set of constraints between the resource throughput and the WIP in a particular enterprise j:

$$\theta_{rj} \leq k_{rj} \tag{14.7}$$

$$w_{rj} \geq \alpha^1_{rj}\theta_{rj} \tag{14.8}$$

$$w_{rj} \geq \alpha^2_{rj}\left(\theta_{rj} - k^0_{rj}\right) \tag{14.9}$$

Now, when considering the set of enterprises $\mathcal{N}$, it can be assumed that slope parameters α^1_{rj} and α^2_{rj} characterize resource r at any enterprise $j \in \mathcal{N}$. Only capacity parameters k_{rj} and k^0_{rj} depend on the considered firm $j \in \mathcal{N}$. Then, in a subset of enterprises $S \subseteq \mathcal{N}$, constraints related to resources of type r owned by the enterprises in S take the following aggregated form:

$$\theta_r = \sum_{j=1}^{N} \theta_{rj} \cdot (e_S)_j \leq \sum_{j=1}^{N} k_{rj} \cdot (e_S)_j \tag{14.10}$$

$$w_r \geq \alpha^1_r \theta_r \tag{14.11}$$

$$w_r \geq \alpha^2_r \left(\theta_r - \sum_{j=1}^{N} k^0_{rj} \cdot (e_S)_j\right) \tag{14.12}$$

Constraint (14.10) is the standard capacity constraint included in the set of constraints (14.5) and related to throughputs θ_{rj} computed by (14.6). Constraints (14.11) and (14.12) express that the aggregated WIP of resource r is above the inverse clearing function. Parameters α^1_r and α^2_r characterize the resource r owned by enterprises in the network. They are both positive and satisfy for all $\alpha^1_r < \alpha^2_r$. Capacity parameters k_{rj} and k^0_{rj} satisfy $0 \leq k^0_{rj} < k_{rj}$.

Equality in Constraint (14.11) or (14.12) will be obtained through constraint saturation when minimizing w_r in the criterion, as it will be the case when minimizing the holding costs.

Using models (14.10)–(14.12), for all the resources, we define the following arrays of resource-dependent parameters:

- the two non negative vectors $\left[\alpha_1^1 \cdots \alpha_R^1\right]^T$, $\left[\alpha_1^2 \cdots \alpha_R^2\right]^T$
- matrix $K_0 = ((k_{rj}^0))$of dimension $R \times N$
- the diagonal matrices of dimension $R \times R$ A_1 and A_2 such that $(A_1)_{rr} = \alpha_r^1$, $(A_2)_{rr} = \alpha_r^2$.

14.3 The Piecewise Linear Supply Chain Game

14.3.1 The Profit Maximization Problem

Unit purchasing costs of primary products and manufacturing costs of intermediate and end-products are supposed fixed and given. They are noted c_i for $i \in \{1,\ldots,n\}$ and in vector form $c = [c_1,\ldots,c_n]^T$. The unit holding cost of workload w_r of resource r is denoted h_r. In vector form, $h = [h_1 \quad \cdots \quad h_R]^T$.

End-products are the goods sold on the market at fixed and given market prices: p_i for $i \in \{1,\ldots,g\}$, and $p = [p_1,\ldots,p_n]^T$ by convention $p_i = 0$ for $i \in \{g+1,\ldots,n\}$.

The profit expected from manufacturing and sale on the market of the vector of outputs y is given by: $(p^T y - c^T \omega))$, which can be rewritten as $\left(p^T - c^T (I - \Pi)^{-1}\right) y$. Additional storage costs will be subtracted from this expression to formulate the total expected profit of the chain.

The profit maximization problem related to the possible supply chain formed by the enterprises in $S \subseteq \mathrm{N}$ can then be stated as follows:

$$\begin{cases} \text{Maximize} \quad v = \left[p^T - c^T (I - \Pi)^{-1}\right] y - h^T w \\ \text{subject to} \\ M(I - \Pi)^{-1} y \leq K e_S \\ A_1 M (I - \Pi)^{-1} y - w \leq 0 \\ A_2 M (I - \Pi)^{-1} y - w \leq A_2 K_0 e_S \\ y \in \Re_+^n, W \in \Re_+^R, \quad e_S \in \{0,1\}^N \end{cases} \qquad (P_S)$$

In the cooperative game theory framework, each subset of enterprises $S \subseteq \mathcal{N}$ is considered as a possible coalition of players, with value function $v(S)$ computed as the optimal criterion of the profit maximization problem (P_S). The set of problems (P_S) over all the possible coalitions $S \subseteq \mathcal{N}$ defines a cooperative game, called the Piecewise Linear Supply Chain Game, or PLSCG.

It can be noticed that any coalition S' such that $S \subseteq S'$ satisfies $e_S \leq e_{S'}$. Therefore, the optimal solution of (P_S) is feasible for $(P_{S'})$ and, as a consequence,

$v(S') \geq v(S)$. The game $(\mathcal{N}, v)$ is said to be monotonic (in the sense of Granot and Maschler 1998) and the following property derives from monotonicity.

Property 14.1 *The maximal profit v^* that can be obtained by any coalition $S \subseteq \mathcal{N}$ is also obtained by the grand coalition $\mathcal{N}: \ v^* = v(\mathcal{N})$.*

According to Property 14.1, the maximum achievable profit with respect to any coalition $S \subseteq \mathcal{N}$ can be directly obtained by solving problem (P_N) for which $e_{\mathcal{N}} = 1_N$. This property is important for two reasons. The first is that it allows for an easy computation of the maximal possible profit value by Linear Programming. The second is that when vector e_S is given, problem (P_S) only contains continuous variable and the optimal value of its criterion is also the optimal value of the criterion of its dual problem, without any "duality gap." This property will be used to compute a profit allocation policy for the enterprises in the network.

14.3.2 Characterization of Core Allocations

The key problem in forming a supply chain is to guarantee that it will be stable in the sense that no set of member enterprises will prefer to separate from the others to create a more profitable chain. In the cooperative game theory, such a stability property is related to the rationality of the profit allocation policy. The set of efficient and rational profit allocation policies is called the "core" of the game. A profit allocation is noted as$(u_i)_{i \in \{1,\ldots,N\}}$. By definition, it belongs to the core of the game ($\mathcal{N}$, v) if and only if it satisfies the following properties:

Efficiency:

$$\sum_{i \in \mathcal{N}} u_i = v^* \tag{14.13}$$

Rationality:

$$\sum_{i \in S} u_i \geq v(S) \quad \forall S \subseteq \mathcal{N} \tag{14.14}$$

Condition (14.14) indicates that for any subset of player $S \subset \mathcal{N}$, there is no coalition alternative in which he could obtain a strictly greater reward.

14.3.3 The Particular Case of Linear Supply Chain Games

If holding costs were neglected, problem *(P_S)* could be reformulated, as in (Hennet and Mahjoub 2010), to represent the game as a linear supply chain game (LSCG):

$$\begin{array}{c}\text{Maximize} \quad v = g^T y \\ \text{subject to} \quad \text{A}y \leq Ke_S \qquad (Ps') \\ y \in \Re_+^n, \quad e_S \in \{0,1\}^N\end{array}$$

with $g = p^T - c^T(I - \Pi)^{-1}$, $A = M(I - \Pi)^{-1}$.

Linear supply chain games are particular instances of the linear production game (LPG), introduced by Shapley and Shubik (1972), and also studied by Owen (1975) and Van Gellekom et al. (2000). The solution rule proposed by Owen (1975) and called the Owen set, is constructed from the optimal solution of the dual $(D_{N'})$ of problem $(P_{N'})$. The optimal dual variables are interpreted as the marginal costs (or shadow prices) of resources. In an Owen assignment, the payoff of each player equals the value of his resource bundle under the unit marginal cost of resources. Moreover, it has been shown in (Owen 1975) that this vector of payoffs forms a subset of the core in this production game.

A similar construction can be achieved with problem (P_N), with a detailed evaluation of the cost of resources resulting from the introduction of the WIP in the model.

14.3.4 A Profit Sharing Mechanism for the Member Enterprises

Three types of dual variables characterize the dual (D_N) of problem (P_N): variables z_r associated with the first set of constraints in (P_N), variables η_r^1 and η_r^2 associated with the second and third sets of constraints in (P_N). To obtain a more compact formulation of the dual problem, the following vectors of variables are introduced:

$$z = (z_1, \ldots, z_R)^T, \; \eta_1 = (\eta_1^1, \ldots, \eta_R^1)^T, \; \eta_2 = (\eta_1^2, \ldots, \eta_R^2)^T.$$

Problem (D_N) is stated as follows:

$$\begin{array}{c}\text{Minimize} \quad \varphi = q^T z + K_0^T A_2 \eta_2 \\ \text{subject to} \\ (I - \Pi)^{-T} M^T (z + A_1 \eta_1 + A_2 \eta_2 \geq p - \; (I - \Pi)^{-T} c \qquad (D_N) \\ \eta_1 + \eta_2 \leq h \\ z \in \Re_+^R, \; \eta_1 \in \Re_+^R, \; \eta_2 \in \Re_+^R.\end{array}$$

The coefficient of variable z_r in the objective function corresponds to the quantity of resource r available for production if the network,

$$q = (q_1, \ldots, q_R)^T \; \textit{with} \quad q_r = \sum_{j=1}^{N} k_{rj}. \tag{14.15}$$

By analogy to the Owen set for LPG, a profit allocation policy $(u_i)_{i\in\{1,\ldots,N\}}$ defining the profit allocation vector $u = [u_1, \ldots, u_N]^T$ can be constructed from the optimal solution $(z^*, \eta_1^*, \eta_2^*)$ of problem (D_N), in the form:

$$u_j = \sum_{r=1}^{R} \left(k_{rj} z_r^* + \alpha_r^2 k_{rj}^0 \eta_r^{2^*} \right) j = 1, \ldots, N \text{ for} \tag{14.16}$$

By construction, $\sum_{j=1}^{N} u_j = \varphi^*$ and from the strong duality property, $v^* = \varphi^*$. The proposed allocation policy is efficient in the sense of (14.16).

Rationality of this policy can also be shown, as for the Owen set, from the property that the constraints defining the dual problem of (P_S) with e_S fixed, are the same for any coalition S (Van Gellekom et al. 2000).

Then, from the definition of the core, the following property is derived.

Property 14.2 *The feasible payoff* profile $(u_i^*)_{i\in N}$ *defined by relations* (14.16)*belongs to the core of the piecewise linear supply chain game.*

The proposed profit allocation mechanism for the PLSCG has the same property of coalitional stability as the Owen set for the LPG. However, it has been observed that the Owen set solution of the LPG has the drawback of being unfair by not rewarding some enterprises having a positive marginal contribution to the global profit (Hennet and Mahjoub 2010).

Typically, this unfairness mainly arises from the fact that in the optimal dual solution of the dual (D'_N) of the LSCG problem (P'_N), resources in excess have null shadow prices; $z_r^* = 0$ if resource r is in excess in the optimal production plan. Introduction of the piecewise linear resource saturation mechanism in the model has generated an additional term η_r^2 related to the WIP of resource r. One of the purposes in introducing the WIP terms in the model is to improve the fairness of the purely competitive profit allocation policy, which is known to belong to the core of the game. This property will be further studied in the next section and illustrated on a numerical example.

14.4 Coalitional Stability and Fairness

There are several definitions of fairness in cooperative games. The most widely accepted one is characterized by the "balanced contribution property" (Myerson 1977). This property considers all the partial contributions of each player in all possible coalitions $S \subseteq \mathcal{N}$. Because of this exhaustiveness, this approach can be said to rely on the "absolute" value of the players' contribution, measured by the "Shapley value." But fairness may also be a matter of comparison between the payoffs of different players. It then relies on some relative measure of contributions, as described in (Fehr and Schmidt 1999). In Linear Production Games, the competitive allocation policy defined by the Owen set (Owen 1975) is rational.

It belongs to the core of the game. It could also be considered fair if one uses the marginal contribution in $\mathcal{N}$ as a measure of fairness. However, such an allocation policy can be seen as typically unfair with respect to the measures of fairness that rely on absolute or relative contributions. The proposed PLSCG model can be seen as an attempt to conciliate marginal and absolute contributions by introducing the amounts of resources used in the form of WIP costs in the Owen set allocation rule (14.16).

In supply networks, firms strive to use their resources in the most efficient manner in order to increase the value of their output products, which are further transformed along the chain and (or) subsequently sold on the market. In such conditions, fair rewards should be related to the ratio between the added value brought by the firm and the total added value of the chain. In mathematical terms, the partial contribution of a player i to coalition $S \subseteq \mathcal{N}$ with $j \notin S$ is defined by:

$$\Delta_i(S) = v(S \cup \{i\}) - v(S) \tag{14.17}$$

A classical characterization of fairness is the "balanced contributions property" defined in Myerson (1977). A profit allocation policy, denoted as $u_j(\mathcal{N}, v)$, is said to be fair in $\mathcal{N}$ if and only if it satisfies the balanced contributions property:

$$u_j(\mathcal{N}, v) - u_j\left(\mathcal{N} - \{k\}, v^{\mathcal{N}-\{k\}}\right) = u_k(\mathcal{N}, v) - u_k\left(\mathcal{N} - \{j\}, v^{\mathcal{N}-\{j\}}\right) \quad \forall j \in \mathcal{N},\ \forall k \in \mathcal{N} \tag{14.18}$$

where by definition, for all $S \subseteq \mathcal{N}$, $(S, v(S))$ is the subgame of $(\mathcal{N}, v)$ defined by $v(T) = v(S) \quad \forall T \subseteq S$.

A particular allocation policy, called the Shapley value (Shapley 1953) has been shown to be the only policy in $\mathcal{N}$ that possesses the balanced contributions property. It is defined by:

$$u_j(\mathcal{N}, v) = \frac{1}{N!} \sum_{q \in Q} \delta_j(S_j(q)) \tag{14.19}$$

for each j in $\mathcal{N}$, where Q is the set of all $N!$ orderings of $\mathcal{N}$, and $S_j(q)$ is the set of players preceding j in the ordering q.

However, the fairness property in the sense of Myerson (1977) is not a sufficient condition for coalitional stability. The possibility for the Shapley value allocation to also be rational is closely related to the convexity property of the cooperative game. Unfortunately, linear production games are not convex in general (Hennet and Mahjoub 2011) and the same restriction applies to LSCG and PLSCG, which are particular instances or extensions of the LPG.

A valuable alternative to Myerson's definition of fairness can be found in the work of Fehr and Schmidt (1999). They define fairness as inequity aversion or more precisely (and to be consistent with the standard economic approach) as self-centered inequity aversion. Inequity can be measured by the difference between the proposed payoff and a reference payoff, which can be the payoff of another player with a similar contribution. The inequity aversion is said "self-centered" when players are only concerned with negative differences. This approach is consistent with many empirical observations of the importance of relative payoffs, especially in games where players contribute equally to the global output. The difficulty for applying this approach to LSCG is in the heterogeneous nature of the resources owned by the firm and in their contrasted importance in the global manufacturing process.

Another definition of fairness, which can also be seen as a minimal requirement for fairness, has been proposed in Hennet and Mahjoub (2010). Consider the set of winning coalitions defined by: $W = \{S \subseteq \mathcal{N}; v(S) = v^*\}$. This definition of fairness simply requires that all the players who belong to a winning coalition with minimal cardinality should at least receive a strictly positive payoff.

A sufficient condition for this property to hold true for a rational allocation policy has been obtained for the LSCG. However, there are many LSCG instances for which a core policy cannot be fair, even when using this definition of fairness. In particular, the following result holds for the LSCG.

Property 14.3 *If the winning coalition with minimal cardinality of the LSCG is not unique, then any core allocation policy is unfair in the sense of Hennet and Mahjoub* (2010).

Proof Consider the LSCG $\left(P'_S\right)$ with optimal value $v^* = v(\mathcal{N})$ and consider the set of winning coalitions: $W = \{S \subseteq \mathcal{N}; v(S) = v^*\}$. Suppose that the payoff profile $(u_i^*)_{i \in N}$ belongs to the core of the game. If the winning coalition of the LSCG with minimal cardinality is not unique, then, with Property 14.1,

$$\forall S \subset W \text{ with } \mathrm{Card}(S) < n, \exists j \in S \, \exists \mathrm{S'} \subset W \text{ such that } j \notin S'.$$

Monotonicity of the LSCG then implies $S' \cup \{\cdot j\} \in W$ and thus, from the efficiency of the allocation policy,
$v^* = v(S' \cup \{j\}) = u_j + \sum_{i \in S'} u_i = u_j + v^*$, which implies $u_j = 0$.

It is not difficult to construct LSCG instances for which the winning coalition with minimal cardinality is not unique. It is the case, in particular, if 2 players own the same resource and only this resource, which is used but not critical in any winning coalition, then, any one of these 2 players is required but obtains a null reward by any core payoff policy.

One of the advantages of the PLSCG formulation is to provide parameters that may prevent resources from not being rewarded in spite of the fact that they are used in the optimal production.

Consider a resource r used in quantity θ_r in the current solution of problem (P_S) and assume that resource r is not critical from the sole viewpoint of capacity constraint (14.10), that is, $\theta_r < \sum_{j=1}^{N} k_{rj} \cdot (e_S)_j$. Then, by duality, the optimal dual variable associated with constraint (14.10) is $z_r = 0$. However, firm j can receive a strictly positive reward for the use of resource in quantity $\theta_{rj} > 0$ provided that parameters α_r^2, α_r^2 and k_{rj}^0 satisfy condition:

$$0 < k_{rj}^0 < \frac{\alpha_r^2 - \alpha_r^1}{\alpha_r^2} \theta_{rj} \tag{14.20}$$

Under condition (14.20), constraint (14.12) is saturated at the optimal solution, so that the optimal WIP dual variable η_r^2 can take a strictly positive value, generating $u_j > 0$ by relation (14.16).

Then, even if a resource is in excess in the optimal production scheme, it starts getting costly as soon as the WIP starts growing.

The main conclusion of this study is that introduction of the WIP variable in the model and in the criterion increases the fairness of the purely competitive profit allocation policy without compromising stability, since this allocation always belongs to the core of the game.

14.5 An Illustrative Example

Consider a product structure represented by the BOM (Bill of Materials) of Fig. 14.2. The products involved are three final products, numbered 1, 2, 3, two intermediate products (4, 5), and three primary products (6, 7, 8). The three end-products (1, 2, 3) are level 0 products, the two products numbered 4, 5 are level 1 products, and the three products 6, 7, 8 are level 2 products.

Matrix Π associated with the structure of Fig. 14.2 is written as follows:

$$\Pi = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 2 & 0 & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

The vector of unit market prices is $p = [100 \quad 125 \quad 130 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T$. Unit manufacturing costs are $c = [5 \quad 4 \quad 4 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6]^T$. Five resources are necessary for the five products at the different manufacturing stages, with the following resource requirement matrix M,

$$M = \begin{bmatrix} 1 & 0 & 2 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 1 & 0 & 1 \\ 2 & 2 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 2 & 2 & 2 \end{bmatrix}$$

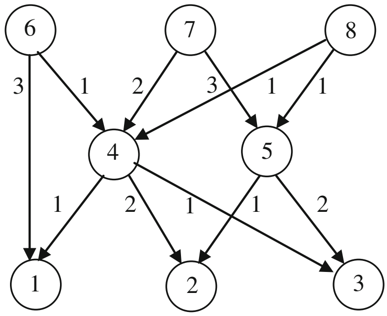


Fig. 14.2 An eight-product structure

Four enterprises are candidate for partnership in the supply chain. The amounts of the five resources owned by the four enterprises are represented in the following matrix:

$$K = \begin{bmatrix} 0 & 100 & 100 & 0 \\ 100 & 0 & 0 & 100 \\ 0 & 20 & 20 & 20 \\ 200 & 0 & 100 & 0 \\ 200 & 100 & 0 & 100 \end{bmatrix}.$$

Unit WIP holding costs are supposed resource dependent only and given by vector: $h = [\,0.5 \quad 1 \quad 0.5 \quad 1 \quad 0.5\,]^T$. Saturation parameters are supposed homogeneous for all the resources and all the enterprises, with $\alpha_r^1 = 0.5$, $\alpha_r^2 = 2$, $k_{rj}^0 = k_{rj}/2$. The optimal total profit is obtained from the solution of the LP (P_N): $v^* = 1115.7$, with $y_1^* = 28.57$, $y_2^* = y_3^* = 0$, $w^* = [85.71 \quad 200 \quad 54.29 \quad 157.14 \quad 400]^T$.

The optimal solution of the dual (D_N) is $\varphi^* = 1115.7$.

It is obtained for $z^* = [0 \quad 0.62 \quad 0 \quad 0 \quad 0.40]^T$, $\eta_1 * = [\,0 \quad 0 \quad 0 \quad 0 \quad 0]^T$ and $\eta_2^* = [\,0.5 \quad 1 \quad 0.5 \quad 1 \quad 0.5\,]^T$.

The profit allocation rule is then computed from expressions (14.16). The unit resource reward vector, denoted rr, takes the value: $rr = [0.50 \quad 1.62 \quad 0.50 \quad 1 \quad 0.90]^T$, and the profit shares of the four enterprises of the supply chain are: $u_1 = 542.86$, $u_2 = 150.44$, $u_3 = 160.00$, $u_4 = 262.41$.

It is observed that all the partners of the chain obtain a strictly positive profit share. If the member enterprises were rewarded only on the basis of the shadow prices of their resources, the profit share of the third enterprise would have dropped to zero. This is because only resources two and five have a strictly positive shadow price and enterprise three does not own any quantity of these resources.

In the proposed allocation rule, any resource r is rewarded not only on the basis of its shadow prices, z_r^*, but also on its utilization cost measured through its WIP marginal cost, η_r^2.

14.6 Conclusions

In this study, we have analyzed a manufacturing network formation problem through the study of a cooperative game named piecewise linear supply chain game (PLSCG). In the setting of the game, the emerging manufacturing network has been modeled as a coalition of partners pooling their resources and sharing the same profit function. The multistage supply chain model has been constructed from the multilevel structure of the manufacturing process. Each manufacturing stage has been identified by its main output product that can either be a component or an end-product. The main originality of the proposed resource model is to take into account not only the capacity and cost of resources, but also the cost of the Work in Process (WIP) generated by their use. For each resource, the WIP is represented by a convex function of the throughput and approximated by a piecewise linear function. The best choice of partners and the best mix of products are then computed by Linear Programming. The optimal solution generates the maximal expected profit for the manufacturing network. Partners of the winning coalition are then supposed to agree on participation in the manufacturing network, provided they receive a profit share that discourages them to break away from the coalition. This is a well-known stability condition verified by any allocation policy that lies in the core of a cooperative game. The difficulty to construct such a core profit allocation policy has been solved by extending to PLSCG the classical Owen set policy defined for linear production games (LPG). In the context of LPG, this policy is known to be perfectly competitive and to reward the players only on the basis of the marginal cost of their resources. However, the use of piecewise linear clearing functions for resources and the introduction of additional variables to represent the workloads have improved the fairness of the Owen profit allocation policy by also rewarding resources that are not critical in the coalition.

Application of the cooperative game approach to manufacturing network formation seems to be very promising since it captures the essence of the game with rather simple models. Many cooperative models of this type can be found in the literature, particularly the ones focusing on distribution and sales rather than manufacturing. There are also many attempts to combine different types of games within the same supply network; in particular cooperative games with close partners and strategic games with distant partners often based on contracts.

References

Asmundsson JM, Rardin RL, Uzsoy R (2003) Tractable nonlinear capacity models for production planning. Research Report, Purdue University (USA)

Baker KR (1993) Requirements planning. In: Graves SC, Rinnooy Kan AHG, Zipkin PH (eds) Handbooks in operations research and management science Vol 4, pp 571–628. North-Holland, Amsterdam

Butera F (1991) La métamorphose de l'organisation: du château au réseau. Les Editions d'Organisation, Paris
Cachon GP, Netessine S (2004) Game theory in supply chain analysis. In: David Simchi-Levi S, Wu D, Shen ZJ (eds) Handbook of quantitative supply chain analysis: modeling in the eBusiness era. Kluwer Publisher, Boston
Fehr E, Schmidt KM (1999) A theory of fairness, competition, and cooperation. Q J Econ 114(3):817–868 (MIT Press)
Gerchak Y, Gupta D (1991) On apportioning costs to customers in centralized continuous review inventory systems. J Oper Manage 10(4):546–551
Granot D, Maschler M (1998) Spanning network games. Int J Game Theor 27:467–500
Hartman BC, Dror M, Shaked M (2000) Cores of inventory centralization games. Games Econ Behav 31:26–49
Hartman BC, Dror M (2003) Optimizing centralized inventory operations in a cooperative game theory setting. IIE Trans Oper Eng 35(3):243–257
Hartman BC, Dror M (2005) Allocations of gains from inventory centralization in newsvendor environments. IIE Trans Sched Logist 37(2):93–107
Hennet JC (2003) A bimodal scheme for multi-stage production and inventory control. Automatica 39(5):793–805
Hennet JC, Mahjoub S (2010) Toward the fair sharing of profit in a supply network formation. Int J Prod Econ 127(1):112–120
Hennet JC, Mahjoub S (2011) Coalitions of firms in manufacturing networks: stability and optimality issues. In: Bittanti S, Cenedese A, Zampieri S (eds) Preprints 18th World IFAC conference, Milano, Italy, pp 6419–6424
Karmarkar US (1993) Manufacturing lead times, order release and capacity loading. In: Graves SC, Rinnooy Kan AHG, Zipkin PH (eds) Handbooks in operations research and management science, vol 4. North-Holland, Amsterdam, pp 287–329
Lamothe J, Hadj-Hamou K, Aldanondo M (2005) Product family and supply chain design. In: Dolgui A, Soldek J, Zaikin O (eds) Supply chain optimisation—product/process design, facility location and flow control. Springer, New York, pp 175–190
Little JDC (1961) A proof for the queueing formula: $L = \lambda W$. Oper Res 9:383–387
Lindner MA, Marquez FG, Chapman C, Clayman S, Hendriksson D (2010) Cloud Supply Chain - A Comprehensive Framework, CloudComp 2010, 2nd International ICST Conference on Cloud Computing, 25-28, Barcelona, Spain
Meca A, Timmer J, Garcia-Jurado I, Borm P (2004) Inventory games. Eur J Oper Res 156(1):127–139
Muller A, Scarsini M, Shaked M (2002) The newsvendor game has a non-empty core. Games Econ Behav 38:118–126
Myerson RB (1977) Graphs and cooperation in games. Math Oper Res 2(3):225–229
Nagarajan M, Sošic G (2008) Game-theoretic analysis of cooperation among supply chain agents: review and extensions. Eur J Oper Res 187(3):719–745
Osborne MJ, Rubinstein A (1994) A course in game theory. MIT Press, Cambridge
Owen G (1975) On the core of linear production games. Math Program 9(1):358–370
Paché G, Bacus-Montfort I (2003) Le management logistique intégré. Problèmes économiques. 2.792
Powell WW, Grodal S (2005) Networks of innovators. In: Fagerberg J, Mowery D, Nelson R (eds) The Oxford handbook of innovation. Oxford University Press, Oxford
Shapley LS (1953) A value for n-person games. Contribution to the theory of games vol II, Kuhn HW, Tucker AW (eds) Annals of mathematics studies 28. Princeton University Press, Princeton
Shapley LS, Shubik M (1972) The assignment game 1: the core. Int J Game Theor 1(1):111–130
Slikker M, Fransoo J, Wouters M (2005) Cooperation between multiple news-vendors with transshipments. Eur J Oper Res 167(2):370–380

Vaszonyi A (1955) The use of mathematics in production and inventory control. Manage Sci 1(1):70–85

Van Gellekom JRG, Potters JAM, Reijnierse JH, Engel MC, Tijs SH (2000) Characterization of the Owen set of linear production processes. Games Econ Behav 32(1):139–156

Williamson O (1981) The modern corporation: origins, evolution, attributes. J. Econ Lit 19:1537–1568

Chapter 15
Stability of Hedonic Coalition Structures: Application to a Supply Chain Game

A. Elomri, Z. Jemai, A. Ghaffari and Y. Dallery

Abstract The goal of this chapter is to provide a study of the coalition formation problem in supply chains using Hedonic cooperative games. The goal is to focus on the problems of (i) coalition structure generation, i.e., formation of coalition structures, such that agents inside a coalition coordinate their activities, but agents of different coalitions will work independently; and (ii) worth sharing, i.e., distribution of the worth generated by the coalition to its agents. We namely demonstrate that when cost-based proportional rule and equal allocation rule are used to divide the total created value, the efficient coalitions always exist and satisfy a set of desirable properties. Further; with the general results, we go deeper into a non-superadditive joint replenishment game with full truckload shipments for which we provide a polynomial algorithmic solution to generate the coalitions

Keywords Coalition stability · Cooperative game theory · Hedonic games · Supply chain management

15.1 Introduction

With advances in information technology, emergence of low-cost outsourcing options in Brazil, Russia, India, and China (BRIC) and other countries, as well as with the growing complexity of end products, supply chains have witnessed a radical transformation from standalone and local supply chains to global

Z. Jemai (✉) · A. Ghaffari · Y. Dallery
Ecole Centrale Paris, Grande Voie des Vignes, 92295 Chatenay-Malabry, Cedex, France
e-mail: zied.jemai@ecp.fr

A. Elomri
Dept. of Mechanical & Industrial Engineering. College of Engineering, Qatar University, 2713 Doha, Qatar

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_15,

networked supply chains in which numerous business entities (suppliers, transporters, manufactures, etc.) belonging to several tiers interact to manufacture highly complex products to customers around the world (Netessine 2009). These networks can enable the creation of superior products at the lowest possible cost while ensuring speedy delivery to the consumer. However, to realize these benefits, a number of new strategies need to be considered. Among these strategies is the formation of alliances. For instance, in spite of working in decentralized structures where each agent is seeking to optimize his profit on his own, supply chain agents may work in coalition structures where the agents within the same coalition align their objectives and coordinate their activities, but agents of different coalitions will work independently or even compete against each other's. These cooperative structures often lead to significant savings. However, to realize these savings, some new challenges need to be addressed. There are two main challenges of the coalition formation problem:

Coalition Structure Generation: this problem concerns the partitioning of supply chain members into disjoint coalitions, such that agents inside a coalition coordinate their activities, but agents of different coalitions will work independently.

Savings Sharing: this problem concerns the distribution of the worth generated by the coalition to its agents, such that each party feels that acting as a coalition is worthwhile for its own sake.

As emphasized by Shenoy (1979), the above issues are closely dependent. On the one hand, the final allocation of payoffs to the cooperating agents depends on the coalitions that they form, and, on the other hand, the formation of coalitions depends on the payoffs available to each agent in each of these coalitions. Therefore, coalition formation and profit allocation should not be addressed sequentially, as it is often the case so far in the literature, but rather simultaneously (Elomri et al. 2012). In this study, we use the principles of hedonic cooperative games to develop a generic procedure that deals with the above cooperative behavior issues in general supply chain settings.

In hedonic games, the outcome of a given actor is totally determined by the identity of the other members of his/her coalition. This class of cooperative games is formally defined by a pair (N, P), where $N = (1, 2, \ldots, n)$ is the set of players, and $P = (\geq_1, \geq_2, \ldots, \geq_n)$ denotes the preference profile, specifying for each player $i \in N$ his/her preference relation $\geq_i$, i.e., a reflexive, complete, and transitive binary relation on set $N_i = (S \subset N : i \in S)$. The main idea of hedonic games is the partitioning of a society into coalitions where each player's payoff is completely determined by the identity of other members of his/her coalition (Bogomolnaia and Jackson 2002; Hajdukova 2006).

In this chapter, we consider the hedonic settings to study the formation of stable coalition structures in inventory games with general cost function. In particular, we consider a set of firms/retailers (players) $N = (1, 2, \ldots, n)$. The firms may form coalitions to achieve some savings. We assume that firms' preference relations are completely determined by the payoff (the portion of savings) that they would gain in each coalition. Therefore, intuitively each player would prefer to join the coalition in which he realizes the largest profit portion. However, this movement will be possible only if it allows the other members of the coalition to be better off. Therefore, given a preference relation, the ideal situation is that each player joins his most preferred coalition. In this case, there will be no reason for one or more players to defect from their coalition. When they exist, these coalitions will be referred to as *efficient coalitions* and will be the core of this study.

As mentioned above, the preference relation in this framework is not exogenous, but rather depending on the profit a player can realize by joining a coalition. Other social and economic considerations that intervene in coalition formation process are out of the scope of this chapter. In other words, the preference relation in our hedonic context will be fully defined by the rule used to divide the savings among cooperating players. In this research, we assume that savings will be allocated proportionally to the contribution of each player. This means that while interacting, the players have the common understanding that the expected outcome of their coalition will be shared proportionally. The use of proportional allocation, in this framework, makes sense from both practical and theoretical perspectives. For instance, game theoretical literature related to proportionality (e.g., Moriarity 1975; Ehud 1977; Roth 1979; Feldman 1999; Ortmann 2000 and Nagarajan et al. 2011) customs proportional allocation as a norm of distributed justice and puts it in the heart of equity theory. Feldman (1999) writes that: “it is the standard of business practice: Profit is typically divided in proportion to investment; and cost is generally allocated on a pro rata basis”.

Under proportional allocation rules, we show that efficient coalition structures always exist in general settings. We provide an algorithm that deals with the generation of such coalitions. Moreover, this efficient coalition structure is shown to be (i) weakly stable in the sense of the coalition structure core and (ii) strongly stable under a given assumption. This framework is then used to study one-supplier multi-retailer Full truckload shipments joint replenishment game (FJR-Game).

The rest of the chapter is organized as follows. Section 15.2 introduces the general model and the associated hedonic game. Section 15.3 describes the formation of stable efficient coalition structures when cost-based proportional allocation and equal allocation are, respectively, employed. Section 15.4 is devoted to the application of our general results to FJR-games. We conclude by summarizing the main insights of our results and discuss some extensions in Sect. 15.5.

15.2 The Model and the Game

In this section, we first present the general joint replenishment model we deal with. We then focus on the introduction of the associated n-person hedonic cooperative game.

15.2.1 The Model

In this chapter, we do not restrict ourselves to a particular supply chain configuration or cost structure. We are developing a general approach that can be applied to any joint replenishment game (JRP−game) as well as to group buying games. In this model, we are given a set of n retailers (for convenience we will use the term firm, player interchangeably with the term retailer), denoted by $N = (1, 2, \ldots, n)$. Retailers place orders to a single supplier to satisfy customer demands. The cost of the optimal inventory replenishment policy of a retailer i working individually (the minimal cost that retailer i can achieve by himself/herself) is denoted $C(i)$. When an alliance is to form, i.e., when a set of retailers, S, decide to cooperate and manage their inventories together by making joint orders, the cost of the optimal inventory replenishment policy of coalition S (the minimal cost that the retailers in S can achieve when they operate jointly without the retailers outside coalition S) is denoted $C(S)$. The incentives to cooperate are not specified here and may include benefits from economies of scale offered by the supplier or/and savings generated by some resources' mutualisation, etc. As mentioned above, we develop here a general approach (the results can be applied not only for JRP-games but also to general cooperative situations). Therefore, the cost function C is a general function that does not need to have special properties like concavity, convexity, or superadditivity.

To evaluate whether a coalition is profitable or not, we need to compare the cooperative situation to the decentralized situation/the standalone situation where each firm is working individually.

To achieve this goal, we let Ω be the space of the $2^n - 1$ possible non-empty coalitions in N and let v a savings function defined as follows:

$$\begin{aligned} & v : \Omega \to R \\ & S \to v(S) = \sum_{i \in S} C(i) - C(S) \end{aligned} \tag{15.1}$$

Definition 15.1 A coalition S is profitable if and only if it has a positive worth $v(S) \geq 0$.

The saving function describes for each coalition of firms, $S \in \Omega$, the maximal worth $v(S)$ that they would divide among themselves if they were to cooperate together and with no firm outside S. The standalone situation where each retailer works on its own constitutes—by construction of function v—the situation of reference. Thus, the worth of single coalition is null, i.e., $i \in N,\ v(i) = 0$.

15.2.2 The Game

Most of this section is based on the papers of Bogomolnaia and Jackson (2002), and Hajdukova (2006). Let P be the finite set of coalition structures. The n-person cooperative game, we are concerned with may be defined by the tuple (N, v, P) where $N = (1, 2, \ldots, n)$ is the set of firms (the players), $P = (S_1, \ldots, S_m)$ is any coalition structure and v the characteristic function of the game is the savings function defined above (Eq. 15.1). We should remember that the savings of a given coalition structure $P = (S_1, \ldots, S_m)$ is the sum of the savings of the coalition's forming it, $v(P) = \sum_{i=1}^{i=m} v(S_i)$. In this model, we consider purely hedonic setting that is each retailer's payoff is completely determined by the identity of the other members of his/her coalition. Formally, each retailer is supposed to have his/her own preferences over coalitions to which he/she could belong. Let us denote by $R = (\geq_1, \geq_2, \ldots, \geq_n)$ the preference profile, specifying for each retailer $i \in N$ his/her preference relation $\geq_i$, i.e., a reflexive, complete, and transitive binary relation on set $N_i = \{S \subseteq N : i \in S\}$. In this model, retailer's preferences are related to the payoff that this retailer will get in each coalition. Thus, if we denote by $\varphi(S, i)$ the expected worth of retailer i in coalition S, $S \in N_i$, asserting that retailer i prefers coalition S to coalition T is equivalent to asserting that his/her corresponding savings is higher in coalition S than in coalition T, i.e.,

$$S, T \in N_i : S \geq_i T \Leftrightarrow \varphi(S, i) \geq \varphi(T, i)$$

Strict preference relations and indifference relations of a player i are, respectively, denoted by $>_i$ and $\sim_i$. Retailer i strictly prefers coalition S to coalition T, means that his/her payoff in coalition S is strictly higher than his/her payoff in coalition T, i.e.,

$$S, T \in N_i : S >_i T \Leftrightarrow \varphi(S, i) > \varphi(T, i)$$

Finally, indifference relations mean that retailer i's payoff is equal in both coalitions S, T, i.e.,

$$S, T \in N_i : S \sim_i T \Leftrightarrow \varphi(S, i) = \varphi(T, i)$$

Formally, the game in coalition structure (N, v, P) may now be defined as a hedonic game defined by the couple (N, R). However, one can wonder how to define the payoff allocation function $\varphi(.,.)$. In keeping with the notions of stability, our aim in what follows is to answer simultaneously cooperative behavior questions of alliance formation and profit allocation. In other words, we are looking for an algorithm which, for any hedonic game (N, R), finds a stable partition. This is equivalent to say that, given a fixed allocation rule, we are looking for an algorithm that builds a stable coalition structure.

15.3 Stable Hedonic Coalition Structures Generation

Given an allocation rule $\varphi(.,.)$, or equivalently a preference profile R, our focus is to study the outcome of the hedonic coalition formation game.

As defined above, firms' preferences are directly related to the profit portion that will be allocated to each firm in each coalition. Thus firm i prefers joining coalition S than coalition T only if its payoff in coalition $(S \cup \{i\})$ is higher than its payoff in coalition $(T \cup \{i\})$, i.e.,

$$S \cup \{i\} \geq {}_i T \cup \{i\} \Leftrightarrow \varphi(S \cup \{i\}, i) \geq \varphi(T \cup \{i\}, i)$$

However, to concretely join coalition S, it is not sufficient that firm i prefers coalition S to T. It is necessary that all the members of coalition S agree to accept firm i. In earlier cooperative game theory works, it was assumed that it is sufficient that at least one member of coalition S is better off and the others are not worst off, to guarantee the acceptance of firm i's membership. In other cases, retailer i's membership will be accomplished even when the members of coalition S are not worst off (each one of them wins at least as much as without player i). In our context, since the firms are independent and the preference criterion is based on the expected worth, we find it reasonable to say that the members of coalition S will accept the membership of firm i only when this will make each one of them "strictly" better off. In this case, coalition S is called a feasible coalition for firm i, and the set of all feasible coalitions for a firm i is denoted by N_i^f.

Definition 15.2 A coalition $S, S \notin N_i$ is *a feasible coalition* for firm i if firm i can join coalition S such feasible move is denoted $i \rightarrow S$. This means that firm i prefers coalition S; at least than staying alone and all the members of coalition S will be strictly better off when firm i joins their coalition. Formally,

$$i \rightarrow S \Leftrightarrow \begin{cases} (S \cup \{i\}) \geq {}_i\{i\} \\ (S \cup \{i\}) > {}_jS, \forall j \in S \end{cases} \tag{15.2}$$

Since each retailer is mainly interested in his/her own profit, it is easy to expect that he/she would like to join the coalition guarantying the maximum worth. In other words, each retailer would like to join his/her most preferred coalition

$$S^{*,i} \in N_i^f \text{ such that } \forall T_i^f, S^{*,i} \geq {}_iT \tag{15.3}$$

Above, we discussed the way the "game" can be played from only one player's (retailer) point of view. Therefore, even if coalition $S^{*,i}$ is the most beneficial (and feasible) coalition for retailer i this do not mean that this coalition will be formed, because obviously coalition $S^{*,i}$ may not be the most preferred coalition for the other (one or more) coalition's members. So the "ideal" is to find a coalition that is the best (the most preferred) coalition simultaneously for all its members, such coalition will be referred to as an efficient coalition.

Definition 15.3 A coalition $S^{*,i}$ is an efficient coalition only when it is the most preferred coalition to each one of its members,

$$S^{*} \text{ is efficient} \Leftrightarrow \in S^{*}, \forall T \in N_i^f; S^{*} \geq_i T \tag{15.4}$$

Proposition 15.1 *Each efficient coalition is "saturated" in the sense that the adhesion of new members is not accepted.*

Proof Considering an efficient coalition S^* and a player j outside $S^*(j \notin S^*)$, by construction coalition S^* is the most preferable coalition for all its members, therefore, $\forall i \in S^*, S^* \geq_i (S^* \cup \{j\})$.

Another interpretation of the above proposition is that when S^* is an efficient coalition, no player outside S^* can join it because this will make him/her worse off or will make coalition S^*'s members worse off. Thus, the worth of each firm inside the efficient coalition is completely dependent on the identity of its other partners without the implication of firms or coalitions outside this coalition.

It is easy to remark that the efficient coalitions are disjoint (by construction). In what follows, the focus will be to address the proprieties of the efficient coalition structure $P^* = \{S_1^*, S_2^*, \ldots, S_m^*\}$ referring to the partition that holds when each firm joins its efficient coalition. However, before moving to this discussion, some natural questions about the efficient coalitions need to be answered. For instance, do the efficient coalitions always exist? If not, what are the conditions under which their existence is guaranteed? And, how to build such coalitions?

The concept of efficient coalitions is defined through players' preferences relations. These preference relations are themselves directly related to the profit portion expected by each player. Therefore, the existence or not of efficient coalitions cannot be studied separately from the allocation rule used to define the preference relations profile. In what follows, we show that efficient coalitions exist at least for two allocation rules; Equal allocations and Proportional allocations. Only both of these allocations are considered over the rest of the chapter.

Definition 15.4 (*Proportional allocations*) The proportional allocation rule distributes the rewards proportionally to the standalone costs. A firm i, member of coalition S will receive the worth

$$\varphi^P(S, i) = \frac{C(i)v(S)}{\sum_{j \in S} C(j)}$$

Definition 15.5 (*Equal allocations*) This allocation rule assigns an equal savings portion to each firm. Thus, a firm i, member of coalition S will receive the worth

$$\varphi^E(S,i) = \frac{v(S)}{|S|}$$

Theorem 15.1 *In a hedonic game* (N, R) *where the preference profile is determined by cost-based proportional allocations or equal allocations,* $\varphi(.,.) \in \{\varphi^P(.,.), \varphi^E(.,.)\}$*, efficient coalitions always exist and are expressed as follows:*

$$\varphi(.,.) \equiv \varphi^P(.,.) : S_1^{*,P} = \arg \max_{S \subseteq N} \left(\frac{v(S)}{\sum_{j \in S} C(j)} \right)$$

$$\varphi(.,.) \equiv \varphi^E(.,.) : S_1^{*,E} = \arg \max_{S \subseteq N} \left(\frac{v(S)}{|S|} \right)$$

Proof The coalition $S_1^{*,P} = \arg\max_{S \subseteq N} \left(\frac{v(S)}{\sum_{j \in S} C(j)} \right)$ always exists by nature of the optimization problem. To show that coalition $S_1^{*,P}$ is an efficient coalition, we should show that any firm in $S_1^{*,P}$ prefers this coalition to any other coalition. So let us consider a firm i and a coalition $T, T \in N_i$, by construction of $S_1^{*,P}$ we have:

$$\frac{v(S_1^{*,P})}{\sum_{j \in S_1^{*,P}} C(j)} \geq \frac{v(T)}{\sum_j C(j)} \Leftrightarrow C(i) \frac{v(S_1^{*,P})}{\sum_{j \in S_1^{*,P}} C(j)} (i) \frac{v(T)}{\sum_{j \in T} C(j)}$$

$$\Leftrightarrow \varphi^P(S_1^{*,P}, i) \geq \varphi^P(T, i) \Leftrightarrow S_1^{*,P} \geq {}_i T, T \in N_i$$

Coalition $S_1^{*,P}$ is then efficient. The proof is similar when we consider equal allocations.

Now let us consider efficient coalition formation under proportional allocations. Once coalition $S_1^{*,P}$ is formed, we suppose that the firms in the new system $N/S_1^{*,P}$ will react similarly, that is the efficient coalition $S_2^{*,P}$ will be formed.

$$S_2^{*,P} = \arg \max_{S \subseteq (N/S_1^{*,P})} \left(\frac{v(S)}{\sum_{j \in S} C(j)} \right)$$

Now the procedure is reapplied; a third efficient coalition $S_3^{*,P}$ will be formed, a fourth and so on until assigning all retailers to their efficient coalitions. It is clear that, by construction, the efficient coalitions are disjoint; therefore, they form a partition of N. This partition refers to **the efficient coalition structure** and is denoted by $P^{*,P}$. When considering, equal allocations, the efficient coalition structure $P^{*,E}$ is formed in the same way as$P^{*,P}$. To summarize, both partitions are formally defined as follows:

Definition 15.6 Efficient Coalition Structures $P^{*,P}$, $P^{*,E}$, refers to the partitions that, respectively, hold when each firm joins its efficient coalition under equal allocations and proportional allocations, i.e., $P^{*,P} = \left\{S_1^{*,P}, S_2^{*,P}, \ldots, S_m^{*,P}\right\}$ and $P^{*,E} = \left\{S_1^{*,E}, S_2^{*,E}, \ldots, S_m^{*,E}\right\}$, such that:

$$S_i^{*,P} = \arg\max_{S \subseteq \left(N / \cup_{j=1}^{j=i-1} S_{ji}^{*,P}\right)} \left(\frac{v(S)}{\sum_{j \in S} C(j)} \right), S_i^{*,P} \in P^{*,P} \tag{15.6}$$

$$S_i^{*,E} = \arg\max_{S \subseteq \left(N / \cup_{j=1}^{j=i-1} S_{ji}^{*,E}\right)} \left(\frac{v(S)}{\sum_{j \in S} C(j)} \right), S_i^{*,E} \in P^{*,E} \tag{15.7}$$

After describing the formation of efficient coalitions, the focus in the rest of the chapter is twofold. First, we analyze the above structures from a cooperative game point of view and secondly, we compare both partitions.

The characterization of efficient coalitions is defined through firms' preference relations. Thus, it may be easy to see that at the individual level each firm will be satisfied to be in its efficient coalition (its most preferred coalition). When extending this analysis to a group of firms, we remark that any subset of firms (within the same efficient coalition) feel that acting in efficient coalition is worthwhile for its own sake and therefore will not defect to form a separate coalition. In keeping with cooperative game theory principles, we can conclude that any efficient coalition is core stable.

Theorem 15.2 *The core of any efficient coalition is non-empty.*

Proof To prove this theorem, we should show that the core of any coalition in $P^{*,P}$, or in $P^{*,E}$ is non-empty. Without loss of generality, let us consider, $S_1^{*,P}$ and $S_1^{*,E}$. Since proportional and equal allocations are imputations, proving the non-emptiness of the core reduces to prove that any subset of firm in an efficient coalition gains at least as much as they can get by themselves if they were to deviate and to form their own coalition.

Let us consider efficient coalition $S_1^{*,P}$. Let T be sub-coalition of $S_1^{*,P}, T \in S_1^{*,P}$ and let us show that $\sum_{i \in T} \varphi^P\left(S_1^{*,P}, i\right) \geq v(T)$.

$$\sum_{i \in T} \varphi^P(S_1^{*,P}, i) = \sum_{i \in T} C(i) \frac{v(S_1^{*,P})}{\sum_{j \in S_1^{*,P}} C(j)} \geq v(T)$$

By construction of $S_1^{*,P}$ we have

$$\frac{v(S_1^{*,P})}{\sum_{j\in S_1^{*,P}} C(j)} \geq \frac{v(T)}{\sum_{j\in T} C(j)}$$

This means that sub-set T will not defect from efficient coalition $S_1^{*,P}$, cost-based allocation, $\varphi^P(S_1^{*,P},.)$ is then a core allocation for the game $(S_1^{*,P},v)$. Similarly, equal allocation, $\varphi^E(S_1^{*,E},.)$ is a core allocation for the game $(S_1^{*,E},v)$.

At this level of our analysis, we only focus on the propriety of an efficient coalition without the implication of firms and the other alliances outside this efficient coalition. However, as one can expect, studying the stability of a coalition structure in general, particularly that of partitions $P^{*,P}$ and $P^{*,E}$, implies the study of the possible interactions between coalitions, i.e., the possible moves of groups of firms that are not only in the same coalition but also belonging to several coalitions.

When dealing with this issue, i.e., the stability of coalition structures $P^{*,P}$ and $P^{*,E}$, the first point to note is that both of coalition structures $P^{*,P}$ and $P^{*,E}$ are weakly stable (this is an immediate result from Theorem 15.2).

Theorem 15.3 (Weak stability) *Efficient coalition structures $P^{*,P}$ and $P^{*,E}$ are weakly stable in the sense that the cost based proportional rule is in the core of any coalition of $P^{*,P}$ and equal allocation is in the core of any coalition of $P^{*,E}$:*

$$\varphi^P(S_k^{*,P},.) \in Co(S_k^{*,P},v) \text{ for all } S_k^{*,P} \in P^{*,P}$$
$$\varphi^P(S_k^{*,E},.) \in Co(S_k^{*,E},v) \text{ for all } S_k^{*,E} \in P^{*,E}$$

The weak stability exposed above means that in the efficient coalitions no group of firms within the same efficient coalition will have the incentive to deviate. When extending this analysis to include the movement of group of firms that may belong to several coalitions, we have the following results.

Theorem 15.4 (Strong stability (stability in the sense of coalition structure core))
1. *Given the cost-based proportional allocation, $\varphi^P(.,.)$, efficient coalition $P^{*,P}$ is a stable coalition* structure.
2. *Given equal allocation rule, $\varphi^E(.,.)$, efficient coalition $P^{*,E}$ is a stable coalition structure.*

Proof The proof of the strong stability is strictly the same for both coalitions structures $P^{*,P}$ and $P^{*,E}$ and is, like in the above theorems, valid by construction. Let us assume that the cost-based proportional rule is the allocation used in the system and let us focus on the stability of $P^{*,P}$. Since we know that the proportional rule ensures the weak stability of $P^{*,P}$, studying the strong stability is equivalent to studying the possible moves of firms that are members of at least two distinct coalitions. Thus, we should show that such sub-coalitions cannot be formed.

Let T be a group of firms members of different coalitions. That is, there exists a set of coalitions $\{S_k^{*,P}, \ldots, S_l^{*,P}\} \subset P^{*,P}$ such that $T \subset \left(\cup_{j=k}^{j=l} S_k^{*,P}\right)$ and $T \cap S_j^{*,P} \neq 0 \quad \forall j \in \{k, \ldots, l\}$. Without loss of generality, we suppose that

$S_k^{*,P}$ is the coalition having the maximum rate value $\frac{v\left(S_k^{*,P}\right)}{\sum_{j \in S_k^{*,P}} C(j)}$ among coalitions $\{S_k^{*,P}, \ldots, S_l^{*,P}\}$. This implies that $\frac{v\left(S_k^{*,P}\right)}{\sum_{j \in S_k^{*,P}} C(j)} \geq \frac{v(T)}{\sum_{j \in T} C(j)}$, otherwise coalition $S_k^{*,P}$ is not satisfying the criteria of efficiency and coalition T would be an efficient coalition, which is not the case. With these introduced proprieties, let us look to the sub-coalition $T' = T \cap S_k^{*,P}$. These firms if they were to deviate from their coalition $S_k^{*,P}$ to coalition T, the worth of each one of them will decrease because,

$\varphi^P\left(S_k^{*,P}, i\right) = C(i) \frac{v\left(S_k^{*,P}\right)}{\sum_{j_k^{*,P}} C(j)} \geq \varphi^P(T, i) = C(i) \frac{v(T)}{\sum_{j \in T} C(j)}$.

Consequentially, coalition T cannot be formed. As mentioned above, the proof is the same when we consider equal allocation.

15.3.1 Complexity Analysis

In this section, our aim is to highlight the computational complexity of generating coalition structures $P^{*,P}$ and $P^{*,E}$. As already stipulated, the formation of coalitions structures $P^{*,P}$ and $P^{*,E}$ implies solving the respective optimization problems (15.6 and 15.7),

$S_i^{*,P} = \arg\ \max_{S \subseteq \left(N / \cup_{j=1}^{j=i-1} S_j^{*,P}\right)} \left\{\frac{v(S)}{\sum_{j \in S} C(j)}\right\}$ and

$S_i^{*,E} = \arg\max_{S \subseteq \left(N / \cup_{j=1}^{j=i-1} S_j^{*,E}\right)} \left\{\frac{v(S)}{|S|}\right\}$.

To find the most efficient coalitions $S_1^{*,P}$ and $S_1^{*,E}$, the space of all possible coalitions is explored. However, in a system of n firms, there are $(2^n - 1)$ possible coalitions. Both problems of generating efficient coalition structures $P^{*,P}$ and $P^{*,E}$ have an exponential complexity. Note, however, that both optimization problems include functions in ratio forms. We advocate then the use of fractional programming theory to tackle the solution complexity (see Sect. 15.4.3).

15.3.2 Comparisons

After discussing the main proprieties of coalition structures $P^{*,P}$ and $P^{*,E}$ from both cooperative game theory and computational points of view, the next natural question to be asked is how to compare these coalitions' structures. We should note that with the use of cooperative game theory the quality of any coalition structure is quite often evaluated through its stability. However, in the current work both coalition structures fulfill the same stability properties. To compare coalition structures $P^{*,P}$ and $P^{*,E}$, we should include more criteria and ask other questions, for instance: Do firms prefer one coalition structure to another? Does one of both partitions contains more coalitions or is more profitable than the other?

As one can expect, it is impossible to answer the above questions in the current general form of the cooperative game. To achieve our goal, we need to apply the afore-described results to an example of supply chain game with an explicit cost structure. This will be our focus in the rest of the chapter. We will consider both scenarios in a one-supplier multi-retailer full truckload shipments joint replenishment game (FJR-game). In the first scenario, the firms will form efficient coalition structure $P^{*,P}$ whereas in the second one coalition structure $P^{*,E}$ will be considered. Since the questions of stability and gains splitting are valid in the general case, we will mainly investigate two topics: (1) The algorithmic question of generating the efficient coalitions, and (2) the comparison of the two scenarios.

15.4 Application: One-supplier Multi-retailer Full Truck Load Shipments Joint Replenishment Game

15.4.1 Model Description and Notations

We consider the issue of generating the afore-studied coalition structures for the $P^{*,P}$ and $P^{*,E}$ for the *single-supplier multi-retailer full truckload shipments joint replenishment game* (*FJR-Game*).

This FJR-Game (Fig. 15.1a) can be stated as follows: A number of independent retail facilities, $N = \{1, \ldots, n\}$, faces known demands, D_i, of a single product—characterized by a volume (or a weight) V_i– over an infinite planning horizon. They order goods from the same external supplier. All shipments from supplier's warehouse to retailers are direct full truckload shipments; all trucks have the same capacity limit called *CAP*.

There are a fixed cost *A*, and a variable cost G_i per truck dispatched from the supplier to retailers, and linear holding costs at the retailers' warehouses. The cost of holding one unit of product per unit of time at retailer *i* is h_i. For simplification, we let $H_i = \frac{h_i D_i}{2}$ be the holding cost parameter of retailer *i*. All costs are stationary costs; i.e., the fixed and variable transportation charges and the linear holding costs do not change over time. Both of transportation costs and linear inventory holding costs involved by products' storage are supported by the retailer.

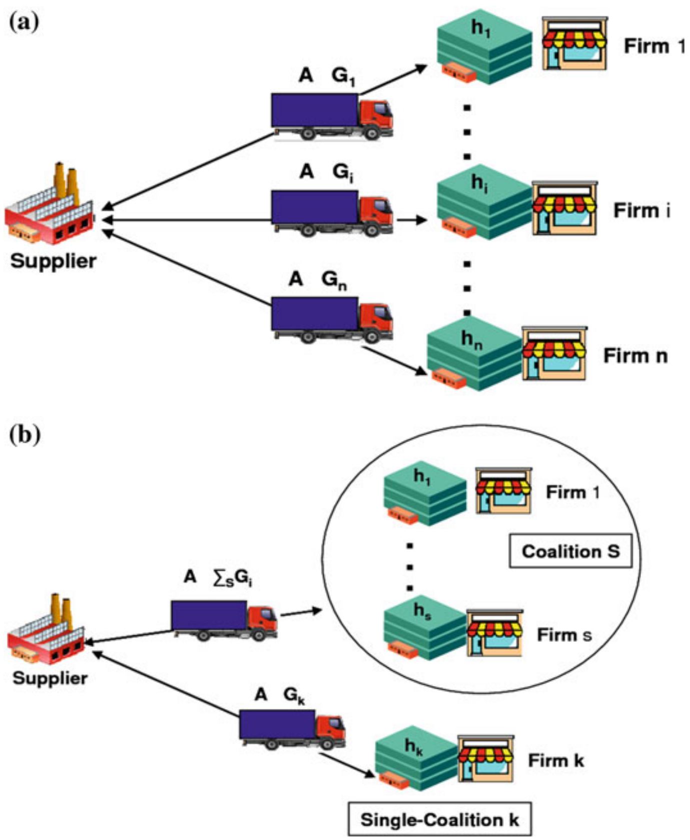


Fig. 15.1 One-supplier multi-retailer full truckload joint replenishment game with three cost components: holding cost, fixed and variable transportation costs, (**a**) standalone situation, (**b**) cooperative situation

15.4.2 One-supplier Multi-retailer Full Truck Load Joint Replenishment Games

Each time a full truckload delivery is requested by a retailer i, a fixed ordering cost A is charged. In addition, a retailer-dependent cost G_i, called individual cost is supported. When a group of retailers form an alliance S, by joining their orders as a single large order, they will pay only one ordering cost A for the full truck shipment while all individual costs will be kept (see Fig. 15.1b). This means that the delivery cost will be $A + \sum_{i \in S} G_i$. When ordering jointly, the common ordering cycle time is denoted by T_S and the corresponding frequency is denoted by N_S. The notations and parameters of the model are summarized below.

- $N = \{1, \ldots, n\}$: The set of retailers;
- D_i: The deterministic demand of retailer $i \in N$;
- G_i: The individual ordering cost of retailer $i \in N$;
- h_i: The holding per unit cost per time unit of retailer $i \in N$;
- V_i: The volume/weight of product i associated to retailer $i \in N$;
- A: The fixed ordering cost;
- CAP: The vehicle capacity;
- T_i: The ordering cycle time of retailer $i \in N$;
- N_i: The ordering frequency of retailer $i \in N$;
- Q_i: The order size of retailer $i \in N$;
- $H_i = \frac{h_i D_i}{2}$: The holding cost parameter of retailer $i \in N$;
- $C(i)$: The average total cost per time unit of retailer $i \in N$;
- T_S: The ordering cycle time of coalition S, $\emptyset \subset S \subseteq N$;
- N_S: The ordering frequency of coalition S, $\emptyset \subset S \subseteq N$;
- $C(S)$: The average total cost per time unit of coalition S, $\emptyset \subset S \subseteq N$.

When ordering alone, the replenishment strategy for a retailer i is to order a full truck corresponding to the quantity $Q_i = \frac{CAP}{V_i}$ every $T_i = \frac{Q_i}{D_i} = \frac{CAP}{V_i D_i}$ unit of time. The corresponding ordering frequency is then: $N_i = \frac{V_i D_i}{CAP}$. Retailer i charges a total delivery cost of $(A + G_i)N_i$ plus a total holding cost $\frac{h_i Q_i}{2}$. Consequentially, the total average cost of retailer i equals $C(i) = (A + G_i)N_i + \frac{h_i Q_i}{2}$. Since $N_i = \frac{D_i}{Q_i}$, rewriting $C(i)$ as a function of the frequency N_i gives :

$$C(i) = (A + G_i)N_i + \frac{H_i}{N_i} \quad \forall i \in \{1, \ldots, n\} \tag{15.8}$$

Above, we have determined the standalone optimal replenishment policy for any firm. In what follows, we focus on the cooperative situation. Consider a non-empty set of firms that decide to form a coalition S to manage their inventory collectively by making joint orders. In this case, it is obvious that in this cooperative structure all these firms will have one common cycle time T_S and a common ordering frequency N_S. Since we suppose that only full truck orders are authorized and no shortage is allowed it is easy to check that the common ordering frequency is the sum of the standalone ordering frequency, i.e.,

$$N_S = \sum_{i \in S} N_i, \quad \emptyset \subset S \subseteq N \tag{15.9}$$

As mentioned above, in the cooperative situation, only one ordering cost is supported. Thus, coalition S charges $(A + \sum_{i \in S} G_i)N_S$ delivery cost. The delivered products are stored in local warehouses where every retailer supports his/her own holding cost; the holding cost charged by the coalition is the sum of the individual holding costs. As a result, the average total cost of alliance S is $C(S) = (A + \sum_{i \in S} G_i)N_S + \frac{\sum_{i \in S} h_i Q_i}{2}$. Expressing the order size Q_i as a function of

N_S leads to: $Q_i = \frac{D_i}{N_S}$. The total average cost of coalition S is then expressed as follows:

$$C(S) = \left(A + \sum_{i \in S} G_i\right) N_S + \frac{\sum_{i \in S} H_i}{N_S} = \left(A + \sum_{i \in S} G_i\right) \sum_{i \in S} N_i + \frac{\sum_{i \in S} H_i}{\sum_{i \in S} N_i} \quad (15.10)$$

Now to discuss whether it is interesting or not for a given set of retailers to cooperate, we should compare the cost in the cooperative situation, $C(S)$ to that in the standalone (decentralized) situation $\sum_{i \in S} C(i)$. As summarized in Table 15.1, on the one hand the cooperative situation leads to a rise in the delivery costs due to the increase of the individual costs charge increase, on the other hand the holding costs in the cooperative situation are lower than the stand alone situation. Profitability is then not guaranteed for all possible coalitions. To be profitable, a given coalition should satisfy the property of proposition 15.2.

Proposition 15.2 *A non-empty coalition $S \subseteq N$ is only profitable when the individual cost raising is balanced by the holding cost decrease.*

$$C(S) \leq \sum_{i \in S} C(i) \Leftrightarrow \sum_{i,j \in S, i \neq j} G_i N_j \leq \sum_{i \in S} \frac{H_i D_i}{N_i} - \frac{\sum_{i \in S} H_i D_i}{\sum_{i \in S} N_i}$$

A direct consequence of proposition 15.2 is that the merging of two or more coalitions into one coalition does not guarantee a total cost decrease. Consequentially, the grand coalition may be non-profitable: The game is non-superadditive.

In the rest of the chapter, the aim is to study the following games with coalition structures: $(N, v, P^{*,P})$ and $(N, v, P^{*,E})$. Where N is the set of firms, v is the savings function.

$$\begin{aligned} &v : \Omega \rightarrow R \\ &S \rightarrow v(S) = \sum_{i \in S} C(i) - C(S) \end{aligned} \quad (15.11)$$

$P^{*,P}$ and $P^{*,E}$ are the efficient coalition structures studied above (see Definition 15.6). $P^{*,P} = \{S_1^{*,P}, S_2^{*,P}, \ldots, S_m^{*,P}\}$ such that:

Table 15.1 Standalone situation versus cooperative situation

	Standalone situation	Cooperative situation	Variation
Delivery costs	$\sum_{i \in S} (A + G_i) N_i$	$\left(A + \sum_{i \in S} G_i\right) \sum_{i \in S} N_i$	↗
Holding costs	$\sum_{i \in S} \frac{H_i}{N_i}$	$\frac{\sum_{i \in S} H_i}{\sum_{i \in S} N_i}$	↘

$$S_i^{*,P} = \arg\max_{S\subseteq\left(N/\cup_{j=1}^{j=i-1} S_j^{*,P}\right)} \left\{\frac{v(S)}{\sum_{j\in S} C(j)}\right\}, S_i^{*,P} \in P^{*,P}$$

and $P^{*,E} = \left\{S_1^{*,E},\ S_2^{*,E},\ \ldots, S_m^{*,E}\right\}$ such that:

$$S_i^{*,E} = \arg\max_{S\subseteq\left(N/\cup_{j=1}^{j=i-1} S_j^{*,E}\right)} \left\{\frac{v(S)}{|S|}\right\}, S_i^{*,E} \in P^{*,E}$$

The rest of the chapter is organized as follows. The first aim is to study the optimization problems of efficient alliance formation. Particularly, we focus on solving the following optimization problems:
$S_1^{*,P} = \arg\ \max_{S\subseteq N}\left\{\frac{v(S)}{\sum_{j\in S} C(j)}\right\}$ and $S_1^{*,E} = \arg\ \max_{S\subseteq N}\left\{\frac{v(S)}{|S|}\right\}$

Once alliance formation problems are solved, a set of numerical tests is provided to compare both coalition structures.

15.4.3 Scenario 1: Coalition Structure $P^{*,\ P}$

The proposal of this section is to provide an exact solution for searching for the efficient coalition structure $P^{*,P}$. As explained above, we will focus on the optimization problem of generating the most efficient coalition:

$$S_1^{*,P} = \arg\ \max_{S\subseteq N}\left\{\frac{v(S)}{\sum_{j\in S} C(j)}\right\} \tag{15.12}$$

Proposition 15.3 *Maximizing the profit ratio is equivalent to minimize the ratio of the coalition‘s cost to its corresponding decentralized cost*:

$$S_1^{*,P} = \arg\ \min_{S\subseteq N}\left\{\frac{C(S)}{\sum_{i\in S} C(i)}\right\} \tag{15.13}$$

Proof

$$S_1^{*,P} = \arg\ \max_{S\subseteq N}\left\{\frac{v(S)}{\sum_{i\in S} C(i)}\right\} = \arg\ \max_{S\subseteq N}\left\{1 - \frac{C(S)}{\sum_{i\in S} C(i)}\right\}$$
$$= \arg\ \min_{S\subseteq N}\left\{\frac{C(S)}{\sum_{i\in S} C(i)}\right\}$$

In what follows, for simplicity, we consider the minimization problem (15.13). The ratio $\frac{C(S)}{\sum_{i \in S} C(i)}$ refers to us as the cost ratio and will be denoted by $CR(S)$.

The optimization problem (15.13) may be formulated as the following linear program. The decisions variables X_j address the selection of one coalition from all possible 2^n-1 coalitions.

$$X_j = \begin{cases} 1: & \text{if coalition } j \text{ is selected} \\ 0: & \text{otherwise} \end{cases}$$

(F.I)

$$\min \sum_{j=1}^{2^n-1} \left(\frac{C(S)}{\sum_{i \in S} C(i)} \right) X_j \tag{15.14}$$

$$\sum_{j=1}^{2^n-1} X_j = 1 \tag{15.15}$$

$$X_j \in \{0, 1\}, \quad \forall j = 1, 2, \ldots, 2^n - 1 \tag{15.16}$$

Because of its exponential complexity, the current *(F.I)* may be only used for systems with a small number of firms. When dealing with a large number, the problem becomes too complex to allow the use of exhaustive enumeration. Since we deal with an objective function that aims at minimizing a ratio of two functions, fractional programming theory may be used to reformulate the problem. To achieve 0-1 fractional program formulation, we define the following new decision variables:

$$Y_i = \begin{cases} 1: & \text{if retailer } i \text{ is in coalition } S \\ 0: & \text{otherwise} \end{cases}$$

Expressing the cost reduction ratio of one coalition S with the newly added decision variables Y_i gives the following 0–1 fractional ratio that represents the objective function (for simplicity $C(i)$) will be replaced by C_i.

$$\begin{aligned} CR(S) &= \frac{\left(A + \sum_{i=1}^n G_i.Y_i\right) \sum_{i=1}^n N_i.Y_i + \frac{\sum_{i=1}^n H_i.Y_i}{\sum_{i=1}^n N_i.Y_i}}{\sum_{i=1}^n C_i.Y_i} \\ &= \frac{A.\sum_{i=1}^n N_i.Y_i}{\sum_{i=1}^n C_i.Y_i} + \frac{\left(\sum_{i=1}^n G_i.Y_i\right)\left(\sum_{i=1}^n N_i.Y_i\right)}{\sum_{i=1}^n C_i.Y_i} + \frac{\sum_{i=1}^n H_i.Y_i}{\left(\sum_{i=1}^n N_i.Y_i\right) \sum_{i=1}^n C_i.Y_i)} \\ &= \frac{A.\sum_{i=1}^n N_i.Y_i}{\sum_{i=1}^n C_i.Y_i} + \frac{\sum_{i=1}^n \sum_{j=1}^n N_i.G_j.Y_i.Y_j}{\sum_{i=1}^n C_i.Y_i} + \frac{\sum_{i=1}^n H_i.Y_i}{\sum_{i=1}^n \sum_{j=1}^n N_i.C_j.Y_i.Y_j} \end{aligned}$$

Formulation (*F.I*) is then equivalent to the following formulation.

(*F.II*)

$$\min \frac{A.\sum_{i=1}^{n} N_i.Y_i}{\sum_{i=1}^{n} C_i.Y_i} + \frac{\sum_{i=1}^{n}\sum_{j=1}^{n} N_i.G_j.Y_i.Y_j}{\sum_{i=1}^{n} C_i.Y_i} + \frac{\sum_{i=1}^{n} H_i.Y_i}{\sum_{i=1}^{n}\sum_{j=1}^{n} N_i.C_j.Y_i.Y_j} \quad (15.17)$$

$$\sum_{i=1}^{n} Y_i \geq 1 \quad (15.18)$$

$$Y_i \in \{0,1\}, \quad \forall i = 1,2,\ldots,n \quad (15.19)$$

The objective function is represented by expression (15.17). The constraint (15.18) ensures that the selected coalition is non-empty. Binary decision variables Y_i are represented by constraints (15.19). In order to linearize the objective function (15.17), let us define two new variables R and T such that:

$T = \frac{1}{\sum_{i=1}^{n} C_i.Y_i}$ and $R = \frac{1}{\sum_{i=1}^{n}\sum_{j=1}^{n} N_i.C_j.Y_i.Y_j}$

This definition is equivalent to:

$\sum_{i=1}^{n} C_i.Y_i.T = 1$ and $\sum_{i=1}^{n}\sum_{j=1}^{n} N_i.C_j.Y_i.Y_j.R = 1$

With the newly introduced variables R and T, formulation *(F.II)* can be rewritten as :

(*F.III*)

$$\min A.\sum_{i=1}^{n} N_i.Y_i.T + \sum_{i=1}^{n}\sum_{j=1}^{n} N_i.G_j.Y_i.Y_j.T + \sum_{i=1}^{n} H_i.Y_i.R$$

$$\sum_{i=1}^{n} Y_i \geq 1$$

$$\sum_{i=1}^{n} C_i.Y_i.T = 1$$

$$\sum_{i=1}^{n}\sum_{j=1}^{n} N_i.C_j.Y_i.Y_j.R = 1$$

$$Y_i \in \{0,1\}, \quad \forall i = 1,2,\ldots,n$$

Next, nonlinear terms $Y_i.R$, $Y_i.T$, $Y_i.Y_j.T$, and $Y_i.Y_j.R$ can be linearized by introducing additional variables T_{ij} and R_{ij}.

(*F.IV*)

$$\min A.\sum_{i=1}^{n} N_i.T_{ii} + \sum_{i=1}^{n}\sum_{j=1}^{n} N_i.G_j.T_{ij} + \sum_{i=1}^{n}\sum_{j=1}^{n} H_i.R_{ii}$$

$$\sum_{i=1}^{n} Y_i \geq 1$$

Table 15.2 Model *(F.IV)*'s complexity

	Variables		Constraints		
	Binary	Continuous	"="	≥/≤	≥ 0
	n	$2n^2+2$	2	$8n^2+1$	$2.n^2$
Total	$2\,n^2+n+2$		$10.n^2+3$		

$$\begin{aligned}&\sum_{i=1}^{n} C_i.T_{ii} = 1 \sum_{i=1}^{n}\sum_{j=1}^{n} N_i.C_j.R_{ij} = 1T - T_{ij} \le (2 - Y_i - Y_j),\\ &\forall i = 1,2,\ldots,n,\ \forall j = 1,2,\ldots,nT_{ij} \le T,\\ &\forall i = 1,2,\ldots,n,\ \forall j = 1,2,\ldots,nT_{ij} \le Y_i,\\ &\forall i = 1,2,\ldots,n,\ \forall j = 1,2,\ldots,nT_{ij} \le Y_j,\\ &\forall i = 1,2,\ldots,n,\ \forall j = 1,2,\ldots,nT_{ij} \ge 0,\\ &\forall i = 1,2,\ldots,n, \forall j = 1,2,\ldots,nR - R_{ij} \le (2 - Y_i - Y_j),\\ &\forall i = 1,2,\ldots,n,\ \forall j = 1,2,\ldots,nR_{ij} \le T,\\ &\forall i = 1,2,\ldots,n,\ \forall j = 1,2,\ldots,nR_{ij} \le Y_i,\\ &\forall i = 1,2,\ldots,n,\ \forall j = 1,2,\ldots,nR_{ij} \le Y_j,\\ &\forall i = 1,2,\ldots,n,\ \forall j = 1,2,\ldots,nR_{ij} \ge 0,\\ &\forall i = 1,2,\ldots,n,\ \forall j = 1,2,\ldots,nY_i \in \{0,1\}, \quad \forall i = 1,2,\ldots,n\end{aligned}$$

As summarized in Table 15.2, the total number of variables in model (*F.IV*) is ($2\ n^2 + n + 2$).

Proposition 15.4 *The exponentially complex optimization* (*F.I*) *problem is equivalent to the polynomial complex optimization problem* (*F.IV*).

15.4.4 Scenario 2: Coalition Structure $P^{*,\,E}$

In this section, we aim at studying the second scenario where the efficient coalition structure $P^{*,E}$ is to form. Similarly to the previous section, the focus will be on the following optimization problem:

$$S_1^{*,E} = \arg\ \max_{S\subseteq N}\left\{\frac{v(S)}{|S|}\right\} \tag{15.20}$$

The optimization problem (15.20) may be formulated as the following linear program. The decisions variables X_j address the selection of one coalition from all possible ($2^n - 1$) coalitions.

$$X_j = \begin{cases} 1: & \text{if coalition } j \text{ is selected} \\ 0: & \text{otherwise} \end{cases}$$

FE.I:

$$\max \sum_{j=1}^{2^n-1} \left(\frac{v(S_j)}{|S_j|} \right) .X_j \tag{15.21}$$

$$\sum_{j=1}^{2^n-1} X_j = 1 \tag{15.22}$$

$$X_j \in \{0, 1\}, \quad \forall j = 1, 2, \ldots, 2^n - 1 \tag{15.23}$$

We use the same technique in the previous section to reformulate the problem into 0-1 fractional program that we linearize in a second time. To achieve the 0-1 fractional program, we define the following new decision variables:

$$Y_i = \begin{cases} 1: & \text{if retailer } i \text{ is in coalition } S \\ 0: & \text{otherwise} \end{cases}$$

Expressing the objective function (15.21) with the newly added decision variables Y_i gives the following result:

$$\frac{v(S)}{|S|} = \frac{\sum_{i=1}^{n} \left(\frac{H_i}{N_i} \right) .Y_i}{\sum_{i=1}^{n} Y_i} - \frac{\sum_{i=1}^{n} H_i.Y_i}{\sum_{i=1}^{n} \sum_{j=1}^{n} N_i.Y_i.Y_j} - \frac{\sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} N_i.G_j.Y_i.Y_j}{\sum_{i=1}^{n} Y_i}$$

Rewriting the problem (*FE.I*) with the new form of the objective function gives: (*FE.II*):

$$\max \frac{\sum_{i=1}^{n} \left(\frac{H_i}{N_i} \right) .Y_i}{\sum_{i=1}^{n} Y_i} - \frac{\sum_{i=1}^{n} H_i.Y_i}{\sum_{i=1}^{n} \sum_{j=1}^{n} N_i.Y_i.Y_j} - \frac{\sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} N_i.G_j.Y_i.Y_j}{\sum_{i=1}^{n} Y_i} \tag{15.24}$$

$$\sum_{i=1}^{n} Y_i \geq 1 \tag{15.25}$$

$$Y_i \in \{0, 1\}, \quad \forall i = 1, 2, \ldots, n \tag{15.26}$$

The objective function is represented by equation (15.24). The constraint (15.25) ensures that the selected coalition is non-empty. The binary decision variables Y_i are represented by constraints (15.26). In order to linearize the objective function (15.24), we define two new variables R and T such that:

$T = \frac{1}{\sum_{i=1}^{n} Y_i}$ and $R = \frac{1}{\sum_{i=1}^{n} \sum_{j=1}^{n} N_i.Y_i.Y_j}$

This definition is equivalent to:

$\sum_{i=1}^{n} Y_i.T = 1$ and $\sum_{i=1}^{n} \sum_{j=1}^{n} N_i.Y_i.Y_j.R = 1$

With the newly introduced variables R and T, formulation (*FE.II*) can be rewritten as:

FE.III:

$$\max \sum_{i=1}^{n} \left(\frac{H_i}{N_i}\right).T.Y_i - \sum_{i=1}^{n} H_i.Y_i.R - \sum_{i=1}^{n} \sum_{j=1,j\neq i}^{n} N_i.G_j.T.Y_i.Y_j$$

$$\sum_{i=1}^{n} Y_i \geq 1$$

$$\sum_{i=1}^{n} Y_i.T = 1$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} N_i.Y_i.Y_j.R = 1$$

$$Y_i \in \{0,1\}, \quad \forall i = 1,2,\ldots,n$$

Next, nonlinear terms $Y_i.R$, $Y_i.T$, $Y_i.Y_j.T$, and $Y_i.Y_j.R$ can be linearized by introducing additional variables T_{ij} and R_{ij}. The resulting linear mixed-integer program is as follows:

FE.IV:

$$\max \sum_{i=1}^{n} \left(\frac{H_i}{N_i}\right).T_{ii} - \sum_{i=1}^{n} H_i.R_{ii} - \sum_{i=1}^{n} \sum_{j=1,j\neq i}^{n} N_i.G_j.T_{ij}$$

$$\sum_{i=1}^{n} Y_i \geq 1$$

$$\sum_{i=1}^{n} T_{ii} = 1$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} N_i.R_{ij} = 1$$

$$T - T_{ij} \leq (2 - Y_i - Y_j), \quad \forall i = 1,2,\ldots,n, \; \forall j = 1,2,\ldots,n$$

$$T_{ij} \leq T, \quad \forall i = 1,2,\ldots,n, \; \forall j = 1,2,\ldots,n$$

$$T_{ij} \leq Y_i, \quad \forall i = 1,2,\ldots,n, \; \forall j = 1,2,\ldots,n$$

$$T_{ij} \leq Y_j, \quad \forall i = 1,2,\ldots,n, \; \forall j = 1,2,\ldots,n$$

$$T_{ij} \geq 0, \quad \forall i = 1,2,\ldots,n, \; \forall j = 1,2,\ldots,n$$

$$R - R_{ij} \leq (2 - Y_i - Y_j), \quad \forall i = 1,2,\ldots,n, \; \forall j = 1,2,\ldots,n$$

$$R_{ij} \leq T, \quad \forall i = 1,2,\ldots,n, \; \forall j = 1,2,\ldots,n$$

$$R_{ij} \leq Y_i, \quad \forall i = 1,2,\ldots,n, \; \forall j = 1,2,\ldots,n$$

$$R_{ij} \leq Y_j, \quad \forall i = 1,2,\ldots,n, \; \forall j = 1,2,\ldots,n$$

$$R_{ij} \geq 0, \quad \forall i = 1,2,\ldots,n, \; \forall j = 1,2,\ldots,n$$

$$Y_i \in \{0,1\}, \quad \forall i = 1,2,\ldots,n$$

Table 15.3 Model (FE.IV)'s complexity

	Variables		Constraints		
	Binary	Continuous	"="	$\geq / \leq$	≥ 0
	n	$2n^2 + 2$	2	$8n^2 + 1$	$2.n^2$
Total	$2\ n^2 + n + 2$		$10.n^2 + 3$		

As summarized in Table 15.3, the total number of variables in model (*FE.IV*) is $(2\ n^2 + n + 2)$.

Proposition 15.5 *The exponentially complex optimization* (*FE.I*) *problem is equivalent to the polynomial complex optimization problem* (*FE.IV*).

15.4.5 Numerical Results and Comparisons

The focus of this section is to compare efficient coalition structures $P^{*,P}$ and $P^{*,E}$. Using a set of numerical tests, we will discuss whether partition $P^{*,P}$ or $P^{*,E}$ differs by more coalitions or by a higher global profit rate.

The game's parameters are generated randomly. For instance, we use uniform distributions U[100; 500], U[0; 100], and U[1; 10] to respectively generate demand rates, D_i, individual costs G_i and holding costs h_i. In these numerical studies, the ordering cost A and the truck capacity CAP were set, respectively, to 100 and 200. We consider the simple case of identical products' volume and set this parameter to $V_i = 1$. The number of firms in the cooperative game was varied in {5, 10, 15} and for each value of n we deal with 10 instances. All computational experiments were performed on a PC with Intel Core 2 CPU of 3 Ghz and RAM of 0.99 GB. All instances were solved using *ILOG OPL Development Studio 5.2* solver with default parameters.

As mentioned above, the comparison between both coalition structures $P^{*,P}$ and $P^{*,E}$ will be done according to two criteria: the global profit ratio $\left(\pi(P) = \frac{v(P)}{\sum_{i \in N} C(i)}, P \in \{P^{*,P}, P^{*,E}\}\right)$ and the number of coalitions in each coalition structure $(|P|, P \in \{P^{*,P}, P^{*,E}\})$. The resulting numerical results are reported in Table 15.4. We should note that $(\Delta(P^{*,P} - P^{*,E}))$ refers to the difference between both coalition structure's criteria. That is, $(\Delta_\pi = \pi(P^{*,P}) - \pi(P^{*,E}))$ and $\left(\Delta_{|P|} = |P^{*,P}| - |P^{*,E}|\right)$.

When analyzing the above numerical results, the first point to note is that coalition structures $P^{*,P}$ and $P^{*,E}$ are closely similar in terms of their global profit rate and number of coalitions. Consequentially, neither $P^{*,P}$ nor $P^{*,E}$ is a strictly dominating coalition structure. The analysis above does not take the individual preferences of firms into account. To determine whether firms prefer one coalition structure to the other one, we should compare firms' worth in both structures.

Table 15.4 Computing results for $P^{*,P}$ versus $P^{*,E}$

Problem size		$P^{*,P}$		$P^{*,E}$		$\Delta(P^{*,P} - P^{*,E})$	
		$\lvert P^{*,P}\rvert$	$\pi(P^{*,P})$ (%)	$\lvert P^{*,E}\rvert$	$\pi(P^{*,E})$ (%)	$\Delta_{\lvert P\rvert}$	Δ_π (%)
n = 5	Max	3	39,56	3	40,75	0	−1,19
	Mean	2,3	28,57	2,6	28,56	−0,3	0,01
	Min	2	15,29	2	17,99	0	−2,7
n = 10	Max	5	39,74	6	37,94	−1	1,8
	Mean	4,7	29,85	5	29,4	−0,3	0,45
	Min	4	19,45	4	21,9	0	−2,45
n = 15	Max	10	46,67	9	48,25	1	−1,58
	Mean	8	26,57	8	26,69	0	−0,12
	Min	5	13,17	6	16,11	−1	−2,94

In the efficient coalition structure $P^{*,P}$, the savings are attributed proportionally to the standalone cost of each firm. As a result, the value attributed to firm i member of coalition $S_k^{*,P}$ is:

$$\varphi^P\left(S_k^{*,P}, i\right) = C(i)\left(\frac{v\left(S_k^{*,P}\right)}{\sum_{j\in S_k^{*,P}} C(j)}\right), \quad S_k^{*,P} \in P^{*,P}$$

The cost-based proportional rule has the interesting property that the firms within the same coalition get the same profit ratio. For instance, the profit rate of a firm i in coalition $S_k^{*,P}$ (the ratio of its allocated value to its standalone cost) is as follows:

$$\pi\left(S_k^{*,P}, i\right) = \frac{\varphi\left(S_k^{*,P}, i\right)}{C(i)} = \left(\frac{v\left(S_k^{*,P}\right)}{\sum_{j\in S_k^{*,P}} C(j)}\right)$$

Unlike $P^{*,P}$, the savings in the efficient coalition structure $P^{*,E}$ are equally divided. It results that the firms within the same coalition gains the same portion of savings (in term of amount).

$$\varphi^E\left(S_k^{*,E}, i\right) = C(i)\left(\frac{v\left(S_k^{*,E}\right)}{\left|S_k^{*,E}\right|}\right), \quad S_k^{*,E} \in P^{*,E}$$

In this case, the profit rate of a firm i member of coalition $S_k^{*,E}$ is as follows:

$$\pi\left(S_k^{*,E}, i\right) = \left(\frac{v\left(S_k^{*,E}\right)}{C(i)\left|S_k^{*,E}\right|}\right)$$

To discuss whether it is better for firms forming an efficient coalition to have the same portion of savings or to have the same profit ratio, we consider in the following a 10-firm cooperative game and we compare the value allocated to each firm in both partitions $P^{*,P}$ and $P^{*,E}$. The firms' parameters are reported in Table 15.5.

Table 15.5 Firms' parameters

Firm $\{i\}$	D_i	G_i	h_i
{1}	534	91	7
{2}	105	90	1
{3}	496	28	8
{4}	355	28	6
{5}	242	7	10
{6}	232	83	9
{7}	533	9	7
{8}	187	52	2
{9}	274	40	3
{10}	287	50	6

Table 15.6 Formation of coalition structures $P^{*,P}$ and $P^{*,E}$ in a 10-firm cooperative game

Coalition structure $P^{*,P}$				
$S_k^{*,P}$	Firms' outcome			
	$\{i\}$	$C(i)$	$\varphi^P(S_k^{*,P}, i)$	$\pi(S_k^{*,P}, i)$ (%)
{4,5,7}	{4}	827,2	393,08	47,52
	{5}	1129,5	536,73	47,52
	{7}	990,49	470,68	47,52
{3,6,10}	{3}	1117,4	350,08	31,33
	{6}	1112,3	348,48	31,33
	{10}	815,25	255,41	31,33
{8,9}	{8}	342,12	54,1	15,81
	{9}	491,8	77,8	15,81
{1}	{1}	1210	0	0
{2}	{2}	199,75	0	0

Coalition structure $P^{*,E}$				
$S_k^{*,E}$	Firms' outcome			
	{i}	$C(i)$	$\varphi^E(S_k^{*,E}, i)$	$\pi(S_k^{*,E}, i)$ (%)
{3,5,7}	{3}	1117,4	508,71	45,52
	{5}	1129,5	508,71	45
	{7}	990,49	508,71	51,4
{4,6,10}	{4}	827,2	311,5	37,65
	{6}	1112,3	311,5	28
	{10}	815,25	311,5	38,2
{1,9}	{1}	1210	102	8,42
	{9}	491,8	102	20,74
{2,8}	{2}	199,75	12,25	6,13
	{8}	342,12	12,25	3,58

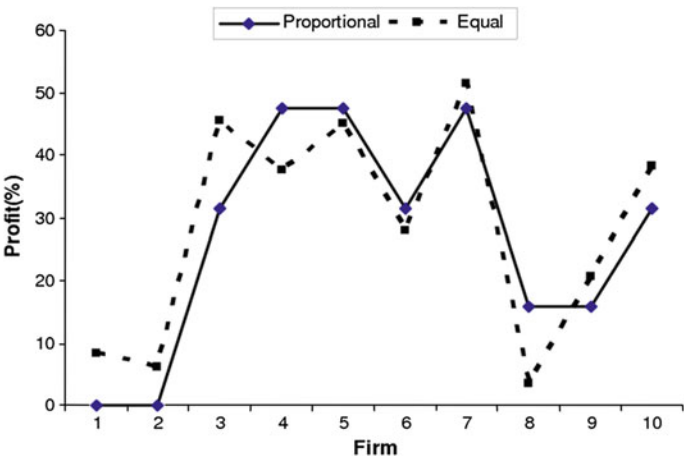


Fig. 15.2 Firms' profit profile in partition $P^{*,P}$ and $P^{*,E}$

The outcome of the game is summarized in Table 15.6. Efficient coalition structure is $P^{*,P} = \{\{4,5,7\},\{3,6,10\},\{8,9\},1,2\}$ and efficient coalition structure is $P^{*,E} = \{\{3,5,7\},\{4,6,10\},\{1,9\},\{2,8\}\}$. In both partitions, coalitions are ranked by their order of formation (efficiency). We reported in Table 15.6, the worth of each firm: we reported the allocated savings portion (in one case the proportional rule is used while in the second case equal allocation is used) as well as the corresponding profit rate. In order to compare firms' created values, we present in Fig. 15.2 the profit rate profile in both coalition structures $P^{*,P}$ and $P^{*,E}$.

When observing the profit rate diagram below, we remark that neither $P^{*,P}$ nor $P^{*,E}$ is strictly better for all the firms. For instance, when moving from one scenario to the other some firms become better off, however, some others become worse off.

To conclude, as discussed above, both coalitions structures $P^{*,P}$and $P^{*,E}$ seem to have the same proprieties. However, we need to be careful when interpreting these results. For instance, despite the apparent "fairness" in coalition structure$P^{*,E}$, the portions of savings are completely independent of the contributions of the cooperating firms. As a result, equal allocation may lead to a situation where those who contribute more are not paid more. This would create "unsatisfied" firms and may thus constitute a motivation for the disbanding of the coalition structure.

15.5 Conclusion and Extensions

In this chapter, we discuss the issue of generating stable coalition structures in games with general cost function. We base our analysis on the principles of hedonic cooperative games. In this theory, the outcome of a given actor is totally determined by the identity of the other members of his/her coalition. Moreover, the formal representation of such games is based on the so called preference profile that specifies for each actor his/her preferences among the coalitions he/she wants to belong to. In this work, we assume that firms' preference relations are linked to the portion of savings that they would gain in each potential coalition. Therefore, each firm would like to join the coalition offering the highest profit portion. Such coalitions, when they exist, are called efficient. Our first contribution was to show that when cost-based proportional rule and equal allocation rule are used to divide the total created value, the efficient coalitions always exist and satisfy a set of desirable properties. For instance, both of efficient coalition structures generated respectively with proportional allocation and equal allocation are stable in the sense of coalition structure core. Further, we stress the exponential complexity of generating such efficient coalition structures.

Our second contribution is to apply the results developed for general models to some concrete joint replenishment games. To achieve this goal, we consider a non-superadditive joint replenishment game with full truckload shipments. Since the questions of forming the efficient coalitions as well as the question of profit allocation are valid in the general case, in the studied FJR-game, we mainly provide a polynomial algorithmic solution to generate the coalitions. Then, using a set of numerical results, we compare both coalition structures. We show that in these games no partition dominates the other. Nevertheless, it's worth noticing that equal savings allocation may lead to "unsatisfied" firms, because such allocation ignores firms' contributions.

Finally, future research on this topic could be aimed at answering the question of whether there exist some other allocation rules that guarantee the existence of efficient coalition structures. We think that the allocations based on marginal contributions such as shapley value and marginal contribution-based proportional allocation do not fulfill this property. We believe that it will be a very interesting contribution to show whether the cost-based proportional allocation and equal allocation are the unique rules that guarantee the existence of efficient coalitions. Another interesting extension of this work is to analyze the issue of considering coalition structures with mixed allocation rules, i.e., the firms express their preference relations according to different allocation rules. We should note that in this model as well as in the theory of games itself, a coalition structure is always assumed to apply the same allocation rule. Even though it will be a difficult problem from the theoretical point of view, we believe that investigating such research direction will provide relevant methods to understand real-world cooperative structures.

References

Bogomolnaia A, Jackson O (2002) The stability of hedonic coalition structures. Games Econ Behav 38:201–230

Ehud K (1977) Proportional solutions to the bargaining situations: interpersonal utility comparisons. Econometrica 45:1623–1630

Elomri A, Ghaffari A, Jemai Z, Dallery Y (2012) Coalitions formation and cost allocation for joint replenishment systems. Prod Oper Manag 21(6):1015–1027

Feldman B (1999) The proportional value of a cooperative game. Manuscript for a contributed paper at the Economic Society World Congress 2000. www.economitricsociety.org/meetings/wc00:pdf/1140.pdf

Hajdukova J (2006) On coalition formation games: a survey. Int Game Theor Rev 8(4):613–641

Moriarity S (1975) Another approach to allocating joint costs. Account Rev 50(4):791–795

Nagarajan M, Sosic G, Zhang H (2011) Stability of group purchasing organizations. Under revision for 2nd-round review at manufacturing and service operation management

Netessine S (2009) Supply webs: managing, organizing and capitalizing on global networks of suppliers. In: Kleindorfer P, Wind Y (eds) Network-based strategies and competencies. Wharton Publishing, New Jersey

Ortmann K (2000) The proportional value for positive cooperative games. Math Method Oper Res 51:235–248

Roth AE (1979) Proportional solutions to the bargaining problem. Econometrica 47:775–778

Shenoy LS (1979) On coalition formation: a game theoretical approach. Int J Game Theor 8:133–164

Chapter 16
Procurement Network Formation: A Cooperative Game Theoretic Approach

T. S. Chandrashekar and Y. Narahari

Abstract In this chapter, we model the multiple unit single item procurement network formation problem as a surplus maximizing network flow cooperative game. Here, each edge is owned by a rational utility maximizing agent. Also, each agent has a capacity constraint on the number of units that he can process. That is, each edge can be assumed to have a capacity constraint on the flow that it can admit. The buyer has a demand for a certain number of units. The agents in the network must coordinate themselves to meet this demand. The buyer also has a specified valuation per unit of the item. The surplus which is the difference between the value generated and the minimum cost flow in the network, is to be divided among the agents that help provide the flow. We first investigate the conditions under which the core of this game is non-empty. We then construct an extensive-form game to implement the core whenever it is non-empty.

Keywords Procurement · Cooperative game · Coordination · Network formation

16.1 Introduction and Motivation

The popular focus of electronic commerce over the last decade has been on developing technologies that facilitate bilateral exchange between two entities such as between a business and a customer (B2C) or between two businesses (B2B). However, complex economic activity often involves inter-relationships at several levels of production, often referred to as supply chains or procurement networks. While a great deal of commercial effort has been devoted to developing technology to maintain pre-existing relationships in the supply chain, precious

T. S. Chandrashekar · Y. Narahari (✉)
Department of Computer Science and Automation, Indian Institute of Science,
Bangalore 560012, India
e-mail: hari@csa.iisc.ernet.in

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8_16,

little has been done to achieve the often expressed vision of virtual organizations where business relationships are dynamically formed and dissolved. This generally requires automated support for a bottom-up assembly of exchange relationships involving complex production and service activities that take place through supplier selection and contracting decisions.

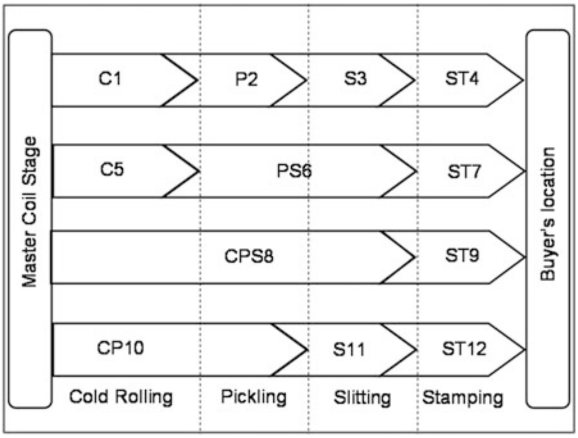
Whether one subscribes to this vision of virtual organizations or not, several business trends already provide evidence that we are moving in this direction. For some examples, we refer the reader to Walsh and Wellman (2003). Complementing these business trends is the increasing tendency by the procurement divisions within organizations to directly get involved in making decisions related to the supplier's supplier in order to plan capacities, monitor incoming quality, and decide prices. For instance, in one of the leading global automotive suppliers that we have worked with, not only do the Original Equipment Manufacturers' (OEMs) explicitly indicate the raw material supplier from whom items need to be sourced but also negotiate raw material prices on behalf of the supplier. This practice of proactively engaging in the formation of the procurement network has been widely adopted by the supplier too. We illustrate the procurement network formation problem, through a stylized example from the automotive industry in the following section.

16.1.1 Procurement Network Formation: What?

Automotive procurement networks typically span many tiers. We consider here the supply chain for an automotive stamping. An automotive assembler, hereafter called the buyer, is interested in procuring stampings for assembly in a automobile. The buyer values the item at a certain price. That is, he associates a maximum value which he is willing to pay to procure the stamping. The stamping undergoes many processes before it can be delivered to the car assembler for assembly in a car. Starting from the master coil, it undergoes cold rolling, pickling, slitting, and stamping before it can be assembled onto the car. We assume that all these manufacturing operations are organized linearly and precedence constraints apply to the way in which the operations can be carried out. This linear supply chain for automotive stampings is shown in Fig. 16.1.

Now, a wide variety of suppliers with varying capabilities may be available in the market to meet the requirements of the assembler. That is, there may be suppliers who are only capable of doing cold rolling while there may be others who can do cold rolling, pickling, and slitting. There may still be others who can deliver the finished stamping by carrying out all the operations. In addition, between each of these manufacturing processes, the item may also have to be transported from one supplier location to another. Each of these suppliers incurs costs to carry out the processing at the various stages of value addition. It is safe to assume that these costs are unique to the particular firm and these costs may be

Fig. 16.1 A linear supply chain for automotive stampings



commonly known or may be privately held information. For the purpose of this chapter and the analysis therein, we make the following assumptions:

- Buyer has complete information about the suppliers' costs.
- Each supplier has complete information about the buyers' valuation and other suppliers' costs.

Note that these assumptions are not too strong because in practice automotive buyers tend to have a handle on every element of the cost involved in the production of most of the components that go into an automobile. The negotiations between the buyer and supplier are therefore centered on agreeing to the terms of quality, delivery, and cost reductions due to process improvements.

Given all these different options in which the stamping can be procured, the buyer has to now decide what is the best (least cost) combination of suppliers he needs to put together to carry out the various operations. This can be done by constructing what we call a procurement feasibility graph which is shown in Fig. 16.2.

In this graph, each edge represents one of the value adding operations which could either be a processing operation or a transportation operation. An edge is assumed to be owned by a supplier, hereafter called an agent. The fact that each agent incurs a certain cost of processing is captured as the cost of allowing a unit amount of flow on the edge owned by him. Also, the processing capacity of the supplier for a particular operation is indicated by an upper bound on the flow that is possible along the edge representing that supplier's operation. In Fig. 16.2, we capture this information as a 3-tuple beside every edge in the graph. Further, we assume that there are two special nodes in the graph called the origin node and terminal node that represent the starting point and ending point for all value adding operations.

A path from the origin node to the terminal node in the graph indicates one possible way of procuring one unit of the stamping. The cost of this procurement is

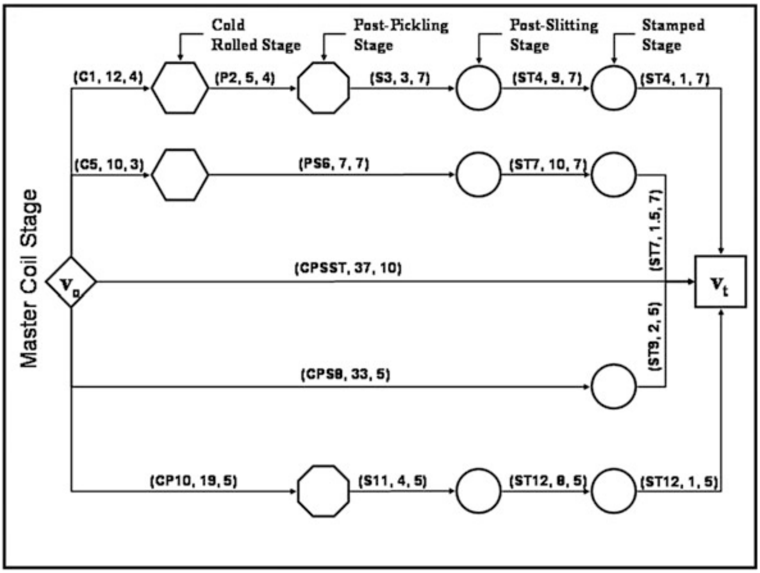


Fig. 16.2 A procurement feasibility graph

obtained by summing the costs of the individual edges in that path. More generally, a flow of f units along a path costs f times the cost of that path. A buyer is then naturally interested in finding a minimum cost way of procuring the stampings to meet his demand or in other words finding a flow through the network that meets his demand and also maximizes the surplus between his valuation of the flow and the cost of maintaining that flow. So, we are essentially interested in a surplus maximizing flow problem which we call as the single item, multi-unit procurement network formation (MPNF) problem.

16.1.2 Procurement Network Formation: How?

The problem of procurement network formation has been variously addressed by researchers under the topic of supply chain formation and hence we use these terms interchangeably in the review of literature. Literature on this topic can be classified into two broad areas—the first where we assume that agents in the network do not act strategically when reporting privately held information and the second where they do act strategically to further their own payoffs. For the first category of problems that are modeled as optimization problems, we point the reader to the survey paper by Erenguc et al. (1999) and for some more recent work to the papers by Garg et al. (2004) and Viswanadham and Goankar (2003). Also see the paper by Biswas and Narahari (2004) for a software implementation of a decision support workbench for supply chains. In this chapter, however, we focus on the second class of problems. Here we assume that agents act strategically. Broadly, the methodological approaches in addressing this problem has followed two tracks: the first uses techniques based on competitive equilibrium analysis such as in Walsh and Wellman (1999, 2003); Erenguc et al. (1999); Viswanadham and Goankar (2003); Garg et al. (2004); Biswas and Narahari (2004) and the second makes use of auction technology as in Walsh et al. (2000); Collins and Gini (2001); Collins (2002); Fan et al. (2003); Jiong and Sadeh (2004); Babaioff and Nisan (2004); Babaioff and Walsh(2005); Garg (2006).

16.1.2.1 Competitive Equilibrium Models

In the competitive equilibrium approach adopted in Walsh and Wellman (1999, 2003); Erenguc et al. (1999); Viswanadham and Goankar (2003); Garg et al. (2004); Biswas and Narahari (2004), the supply chain formation problem is addressed from the perspective of a third party market maker. This market maker is interested in constructing the supply chains of multiple end customers given the fact that they could potentially share some common suppliers. They construct a network of possible supply chain configurations and use a price directed search for a feasible supply chain configuration, i.e., one that maintains material balance and profitability. The price directed search leads to what is essentially an approximate competitive equilibrium which is in line with standard results in competitive equilibrium theory for indivisible goods. They describe two distributed protocols called (Simultaneous Ascending ($M + 1$)st Price with Simple Bidding (SAMP-SB) and SAMP-SB with decommitment (SAMP-SB-D) to find the equilibrium prices along with the supply configuration. To do this, they run a series of simultaneous two-sided auctions for each of the goods to be produced/consumed in the supply chain. The competitive equilibrium that they reach allows them to guarantee that no single active agent in the formed network will find it beneficial to move away from the recommended solution.

This equilibrium-based approach is predicated on the assumption that there are a large number of agents who are a part of the network and hence the scope for strategic behavior by the agents is limited. While it may be true that a large number of suppliers will need to be coordinated along the supply chain, in many real cases, we find that there are a limited number if not just one supplier for particular value adding activities. This leaves ample scope for strategic behavior and hence cannot be assumed away.

16.1.2.2 Concurrent Auction Models

Babaioff and Nisan (2004) consider a linear supply chain where a commodity market exists for each of the goods—both final and intermediate, that can be traded. The supply chain formation protocols that are discussed essentially build on double auction (DA) rules where buyers and sellers submit asks and bids simultaneously. The protocol as such governs the construction of supply and demand curves at each of the markets (supply, intermediate, and final). In each market, after the supply and demand curves are constructed, a double auction rule is invoked to finalize the traded quantities and the prices. The characteristics of the supply chain formation protocol are obtained from the characteristics of the underlying double auction rule that is used in market making. If the DA rule is strategy-proof, efficient, and individually rational but not budget balanced, then so is the supply chain formation protocol. This happens when the DA payment rule is modeled after the Vickrey–Clarke–Groves (VCG) payment rule. With such a rule, the market maker subsidizes the formation of the supply chain. If, however, the DA rule is strategy-proof, budget balanced, and individually rational but is not ex-post efficient, then these characteristics are reflected in the supply chain formation protocol too. This is achieved by using a Trade Reduction (TR) rule (first pointed out by McAfee (1992) or a variant of it. The authors present two randomized DA rules which are essentially randomizations between the VCG rule and the TR rule. With these rules, the supply chain formation protocol is able to achieve ex-post individual rationality and strategy-proofness along with budget balance and suffers from only a slight loss of efficiency. This idea is further extended in Babaioff and Walsh (2005) to more general supply chain structures where two or more goods may be used for making a single good.

16.1.2.3 Combinatorial Auction Models

Another approach to modeling the supply chain formation problem has been to use combinatorial auction/exchange technology. By using combinatorial auction/exchange technology, some of the problems related to uncoordinated actions across the supply chain are avoided; specifically situations where intermediaries are allotted input materials but are not allotted contracts for outputs are avoided (deadlocks). In Walsh et al. (2000), they model the supply chain formation problem assuming single-minded supply chain agents (buyers and sellers). Here single minded refers to the fact that the agents are interested in exactly one particular bundle of goods alone. The agents submit bids for bundles of goods to a central auction/exchange. The auction/exchange then solves a combinatorial optimization problem and indicates the allocations. Without invoking Vickrey payment rules, it is quite evident that the agents would resort to strategic bidding. However, the analysis of bidding behavior to obtain closed-form solutions corresponding to Bayesian-Nash equilibrium turns out to be difficult. So, it is assumed that agents will bid so as to garner a share of the expected surplus. This is done by choosing an averaged bid based on Monte Carlo simulations of the costs of other producers, its own costs, and the valuations of buyers. Experimental results on a variety of networks show that when the surplus is large then the combinatorial auction/exchange technology for supply chain formation outperforms the SAMP-SB protocols. In addition, it has the added advantage of not selecting deadlocked networks.

Fan et al. (2003) model the supply chain formation problem as a multi-commodity flow problem. Standard linear programming techniques such as LP decomposition are the usually used techniques to solve such problems. However, since the cost functions are under the control of agents who solve the decomposed problems, the authors argue that a more natural way to solve the problem is to use a combinatorial auction-based protocol. Here the supply chain agents bid for bundles of goods that they are interested in. The combinatorial auction is constructed to minimize the cost of forming the supply chain while meeting the demands of the customer. The authors make the following key assumptions in their approach to the problem:

- All customer orders are expected to be serviced. In reality, at any given instance, a supply chain wishes to pick the set of orders to be serviced and the required network of supply partners so that the surplus of the supply chain is maximized.
- The resources are considered to be divisible. This enables them to compute prices for individual goods.
- The number of agents is assumed to be very large. This allows them to argue that since the benefits that the agents may expect to get by indulging in strategic bidding is very small, there is very little incentive for them to actually do so and hence they would be better off reporting their true valuations to the auction mechanism. However, most supply chain formation problems in reality need to be addressed within a context where there are only a limited number of agents available for each of the goods that need to be acquired or transformed through the supply chain.

16.1.2.4 Summary of Extant Literature

To summarize the current state of the art and the emerging needs in procurement network formation, we have identified the following key issues:

- In all the literature that we have seen to date, the approach is to solve the procurement network formation problem from the perspective of a third-party market maker. In our view, it would be beneficial to view supply chain formation from the perspective of the buyer/assembler since he is the one who orchestrates the procurement network. Also, this is in line with the current trends in the business environment where companies are moving away from third-party market makers such as COVISINT to setting up their own electronic trading sites.
- Second, most of the literature has focused on using auction-based protocols for dynamic supply chain formation. While this focus has been fruitful in advancing the state of the art for some areas of applications within business-to-business transactions, practitioners have been reluctant to embrace purely auction-based methods across the board.

- In addition, supply chain practitioners and researchers have long argued that a supply chain that is created is sustained when it is able to create a surplus value and the surplus is shared with the supply chain partners. These two points when taken together imply that models for procurement network formation should explicitly include the aspect of surplus sharing.
- In facilitating such a trade, the assumption that agents will bid truthfully because of competitive forces as is assumed in Walsh et al. (2000); Walsh and Wellman (2003); Fan et al. (2003) may be unrealistic to expect since at many levels in the procurement network, we are faced with monopolistic or oligopolistic situations where the agents can gain from indulging in strategic behavior. So, models for procurement network formation must explicitly take into account the possibility of strategic action by suitably engineering the incentives to promote truthful behavior.

Motivated by these gaps in current research and insights into the domain, we believe that the procurement network formation problem may be explicitly analyzed as a surplus sharing network formation problem where we provide for strategic behavior by the agents who own parts of the network. Problems in surplus sharing and cost sharing are replete with ideas from cooperative game theory. We examine next how cooperative game theory is an apt tool to model and solve the procurement network formation problem.

16.1.3 Procurement Network Formation: Our Approach

We have seen from the discussion in the previous section, that auction-based protocols have been widely used to address the dynamic formation of procurement networks. However, there is overwhelming reluctance on the part of buyers to embrace auctions even for settling procurement contracts in the context of trading relationships between agents in adjacent echelons in the supply chain. Both anecdotal evidence and articles in the academic literature seem to overwhelmingly indicate that contractual relationships between agents in a procurement network are characterized by buyers and sellers often bargaining and negotiating over prices, quantities, delivery schedules, and several other attributes. We refer the reader to a recent article by Bajari et al. (2003) that empirically compares the use of auctions versus negotiations in procurement.

Bargaining and negotiation are well-researched topics within the domain of cooperative game theory which, in several recent survey articles and handbook chapters by Bajari et al. (2003); Cachon and Netessine (2004); Wu (2004); Leng and Parlar (2005); Nagarajan and Sosic (2008), has been identified as an important tool in the supply chain researchers toolkit. While non-cooperative game theoretic models have been extensively employed to model the interaction between agents in a supply chain (see Cachon and Netessine 2004), the use of cooperative game theory in the field of supply chain management has, however, been much less prevalent. These approaches, though different in their theoretical content and the methodology used in their analysis, are really two different ways of looking at the same problem. To quote the words of Aumann (1987): the game is one ideal and the cooperative and non-cooperative approaches are two shadows.

The non-cooperative theory of games is strategy oriented, i.e., it studies what one may expect the agents to do in a given interdependent decision-making scenario and the nitty gritty details of the actions that they take to maximize their utility. Cooperative game theory on the other hand takes a different tack. It directly looks at the set of all possible outcomes, studies what agents can achieve, what coalitions will form, how the coalitions that do form divide the spoils of an outcome, and whether the coalitions and outcomes are stable and robust. Thus, one may say that the non-cooperative game theory is a micro approach in that it focuses on the precise details of how things happen in a game theoretic scenario whereas cooperative game theory is a macro approach in that it is concerned with what will happen.

Cooperative game theory has two central themes. The first theme is that of feasible outcomes and the second is that of stability.

- Feasible outcomes represent the set of all possible outcomes that agents may realize with or without the consideration of incentive compatibility constraints. Once this set is identified, then one moves to how agents actually end up with an outcome from this feasible set. Cooperative game theory offers several recipes, known as solution concepts, for this process. The core and the Shapley value are two prominent examples of these solution concepts.
- Stability manifests itself when agents have to decide on allocations of (surplus) value from the set of feasible outcomes. If this is independent of the negotiation process, some or all agents can pursue options such as joining together as a coalition and agreeing upon a joint course of action. Two questions immediately arise: (1) How will agents in an alliance divide the gains that accrue from their joint action? (2) What are the stable coalitions or outcomes that will emerge in a particular scenario? Once again, cooperative game theory offers answers to these questions through its solution concepts.

These two central themes of cooperative game theory have their perfect analogs in the context of procurement network formation. In procurement network formation, we are interested in constructing the feasible set of outcomes and picking the best from among them. In addition, as argued by management scientists, supply chains remain stable only when each of the agents gets a fair share of the surplus that the supply chain generates. So, we are interested in finding a fair way of dividing the gains that accrue when agents come together to create surplus value. In doing so, we are interested in finding a stable coalition of partners. Clearly then, cooperative game theory provides us with an apt toolkit to study the procurement network formation problem.

16.1.4 Contributions and Outline of the Chapter

In this chapter, we focus on the multiple unit, single item procurement network formation problem when the agents who own edges in the procurement feasibility graph are completely aware of (1) each other's costs and (2) the maximum demand of the buyer and his valuation. Our specific contributions are as follows:

- We model the multiple unit, single item procurement network formation problem as a cooperative game with complete information.
- We investigate the conditions under which the core of the MPNF game is non-empty. For this, we first provide illustrative examples of procurement networks to develop intuition and then formalize through analytical results (a) the effect of edge ownership structure on the non-emptiness of the core, (b) the effect of the buyer's valuation on the non-emptiness of the core of a special case of the MPNF game, and (c) the effect of the demanded quantity on the non-emptiness of the core.
- We then develop an extensive-form game to implement the core in sub-game perfect Nash equilibrium, whenever the core is non-empty.

The sequence in which we progress in this chapter is as follows:

In Sect. 16.2, we develop a cooperative game model of the MPNF problem; in Sect. 16.3, we start our investigation of the core as a solution concept for the MPNF game by presenting simple examples to develop intuition about the conditions under which the core of the MPNF game is non-empty. Following this, in Sect. 16.4, we formalize the above intuition. In Sect. 16.5, we address the issue of implementing the core. For this we develop an extensive-form game for the MPNF problem. We then show analytically that the core of the MPNF game in coalitional function form corresponds to the sub-game perfect Nash equilibria of the extensive-form game. Following this, in Sect. 16.6, we present an example to demonstrate the non-cooperative implementation of the core through the extensive-form game developed in Sect. 16.5.

16.2 The Model

As indicated earlier, the feasible network for forming the multiple unit, single item procurement network may be captured as a directed graph. We call this the procurement feasibility graph $G = (V, E)$, with V as the set of vertices, two special nodes V_0 (origin vertex) and V_t (terminal vertex), and $E \subseteq V \times V$ as the set of edges. With each of the edges $e \in E$, we associate the numbers $c(e)$, $l(e)$, and $u(e)$ to represent the cost, the lower bound on the capacity of the edge, and the upper bound on the capacity of the edge, respectively. Now, assume that each of the edges is owned by an agent i where i belongs to a finite set of agents $N = \{1, \ldots, n\}$. We define $\psi : E \rightarrow N$ such that $\psi(e) = i$ implies that agent i owns edge e. We let $I(j)$ and $O(j)$ represent the set of all incoming and outgoing edges at vertex $j \in V$. Note that we allow an agent to own multiple edges.

Let $S \subseteq N$ be a coalition of agents. We let E_S represent the set of edges owned by agents in S. We also designate F_S as the flow in the network between the two special nodes v_0 and v_t using only the edges E_S that are owned by agents in S. The flow on any edge $e \in E_S$ is designated as $f(e)$. For any flow F_S, we denote the set of owners of the edges that facilitate the flow F_S as $\psi(F_S)$. We assume that if multiple units of the item are available to the buyer by using the flow F_S, then it costs $c(F_S)$ and the buyer is willing to compensate the edge owners with a value bF_S where b is the value that the buyer attaches to a single unit of the item. The surplus from such a transaction is $bF_S - c(F_S)$. The maximum demanded quantity of the buyer is d_{v_t}. The problem now is to (a) maximize the surplus as given by the optimization problem below and (b) divide the surplus among the agents in a fair way. These two questions essentially constitute the multiple unit, single item procurement network formation problem. We denote this as $\mu b = (N, G, \psi, b, d_{v_t})$ which in turn induces a cooperative game that can be represented in the characteristic function form as $(N, v_{\mu b})$ where N is the set of agents and $v_{\mu b}$ is the characteristic function given by solving the optimization problem specified by Eqs. (16.1)–(16.5) for every $S \subseteq N$. We are now interested in finding solutions to this game.

$$v_{\mu b}(S) = \max \left[b x_{vt} - \sum_{e \in E_S} c(e) f(e) \right] \tag{16.1}$$

subject to:

$$\sum_{e \in I(j) \cap E_S} f(e) - \sum_{e \in O(j) \cap E_S} f(e) = 0, \forall j \in N \backslash \{v_0, v_t\} \tag{16.2}$$

$$\sum_{e \in I(v_t) \cap E_S} f(e) = x_{v_t} \tag{16.3}$$

$$\sum_{e \in O(v)_0) \cap E_S} f(e) = x_{v_t} \tag{16.4}$$

$$0 \le x_{v_t} \le d_{v_t}, \text{ and } l(e) \le f(e) \le u(e), \forall e \in E_S \tag{16.5}$$

16.3 The Core of the MPNF Game: Key Issues

The core as a solution concept for cooperative games occupies a central place. The notion of the core as consisting of those allocations of the surplus such that they are immune to recontracting among the agents in a cooperative scenario is especially attractive from the perspective of coalitional stability when forming procurement networks. In a practical sense, it signifies that no subset of agents will find it profitable to withdraw from a coalition when the surplus is divided among members of the coalition in a way that it is in the core of the game. However, we know that the core is not always non-empty (see Myerson 1991). Therefore, investigating the non-emptiness of the core of the MPNF game is an important step in addressing the procurement network formation problem.

Formally, it means that we make allocations of the surplus to agents such that they obey Eq. (16.6) where x is an allocation vector and $C(v_{\mu_b})$ the allocations in the core.

$$C(v_{\mu_b}) = \left\{ x \in \Re^n | \sum_{t \in S} x_t \geq v_{\mu_b}(S), \forall S \subseteq N \sum_{t \in N} x_t = v_{\mu_b}(N) \right\} \quad (16.6)$$

The key questions that we need to address regarding the core of the MPNF game are:

1. Does the MPNF game always have a non-empty core?
2. If not, is it at least non-empty under some conditions that can be characterized?

The concept of the core is derived axiomatically and hence does not provide any explicit ideas on the interactions between individual agents or coalitions of agents that yield the final outcomes. If we believe that agents play strategically (non-cooperatively), then the process through which the allocations of the surplus that are in the core of the game emerge is not clear. So, we need to formulate a non-cooperative framework that supports the outcomes that are in the core of the MPNF game. That is, very simply, we need to address the following question: Can we develop an extensive-form game that implements the core outcomes? We address each of these questions in the sections that follow.

First, we would like to know if the MPNF game always has a non-empty core. The short answer to this question is NO! The MPNF scenario is specified by the tuple $\mu b = (N, G, \psi, b, d_{v_t})$. Given the procurement feasibility graph G and the set of agents N who own edges in G, there are three other parameters of the problem: (a) the ownership structure given by $\psi : E \rightarrow N$, (b) the valuation of the buyer b for each unit of the item, and (c) the maximum demanded quantity d_{v_t} of the buyer. We would like to investigate if each of these parameters has any influence on the emptiness or the non-emptiness of the core of the MPNF game. We develop our intuition about this influence by examining some simple examples of procurement networks.

16.3.1 Effect of Ownership Structure on the Core of the MPNF Game

Consider the procurement scenarios given by the procurement feasibility graphs in Figs. 16.3 and 16.4. In both these scenarios, we have three agents—1, 2, and 3 who

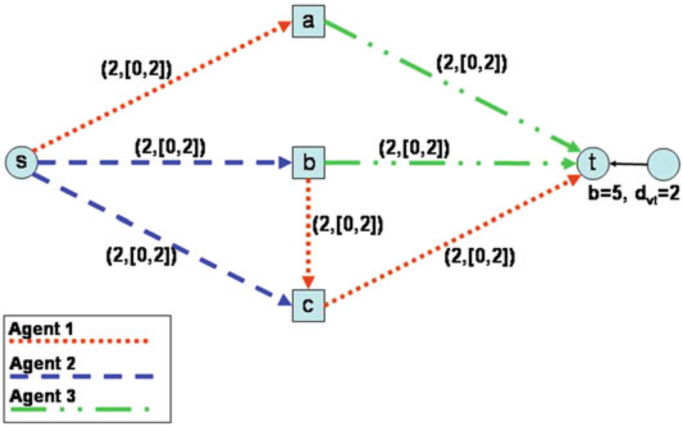


Fig. 16.3 Procurement graph with an empty core

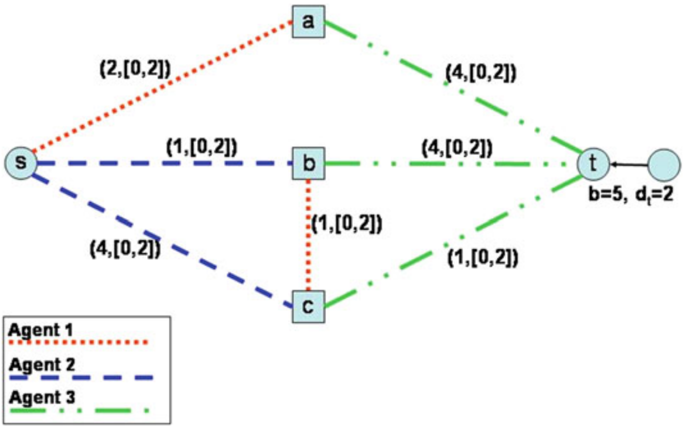


Fig. 16.4 Procurement graph with a non-empty core

own edges in the feasibility graphs. The ownership is indicated by the differently hatched line styles on the edges. A tuple (c, [l, u]) indicated alongside an edge gives us the cost c per unit amount of flow along the edge, a lower bound l, and an upper bound u on the edge capacity. The numbers alongside the labels b and d_{v_t} near the terminal node t indicate the valuation of the buyer for each unit of the item and the maximum demanded quantity, respectively.

From this basic data, it is possible to calculate the maximum surplus $v_{\mu_b}(S)$, that any coalition $S \subseteq N = \{1, 2, 3\}$ can generate on its own, by solving the optimization problem specified by Eqs. (16.1)–(16.5). This generates the characteristic function values for the MPNF game (N, v_{μ_b})which is indicated in Tables 16.1 and 16.2, respectively.

Table 16.1 Characteristic function for the feasible procurement graph in Fig. 16.3

S	Surplus maximum flow F_S by S for $d_{v_t} = 2$	C_S	$v_{\mu_b}(S)$
{1}	0	–	0
{2}	0	–	0
{3}	0	–	0
{1, 2}	2	8	2
{2, 3}	2	8	2
{1, 3}	2	8	2
{1, 2, 3}	2	8	2

Table 16.2 Characteristic function for the feasible procurement graph in Fig. 16.4

S	Surplus maximum flow F_S by S for $d_{v_t} = 2$	C_S	$v_{\mu_b}(S)$
{1}	0	–	0
{2}	0	–	0
{3}	0	–	0
{1, 2}	0	–	0
{2, 3}	1	5	0
{1, 3}	0	0	0
{1, 2, 3}	2	6	4

From an inspection of Table 16.1 corresponding to the Fig. 16.3, it is clear that the core of the MPNF game in this case in empty. That is, there is no allocation of surplus that allows the formation of a stable network with no opportunities for recontracting. Observe that this is because each pair of agents can generate by itself the surplus that the grand coalition (of all agents) can generate.

Now, consider Fig. 16.4 and the corresponding characteristic function values in Table 16.2. Observe that there is a change in the ownership structure of the edges in the graph and hence a corresponding change in the characteristic function. From an inspection of the characteristic function values it is clear that this feasibility graph admits an allocation that is in the core of the corresponding MPNF game. From the discussion of these two procurement graphs, we can conjecture that the non-emptiness of the core of the MPNF game depends on the ownership structure of the edges in the network.

16.3.2 Effect of Buyer's Valuation on the Core of the MPNF Game

To study this, consider the procurement graphs shown in Figs. 16.5 and 16.6. Here, we have only two agents—1 and 2 who own edges in the feasibility graph. The ownership structure and the basic data of the problem are indicated in the figures as before. As before, we can develop the characteristic function values by solving

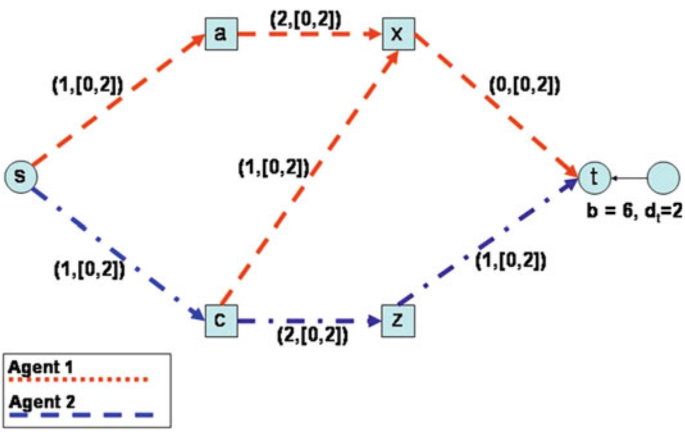


Fig. 16.5 Procurement graph with an empty core

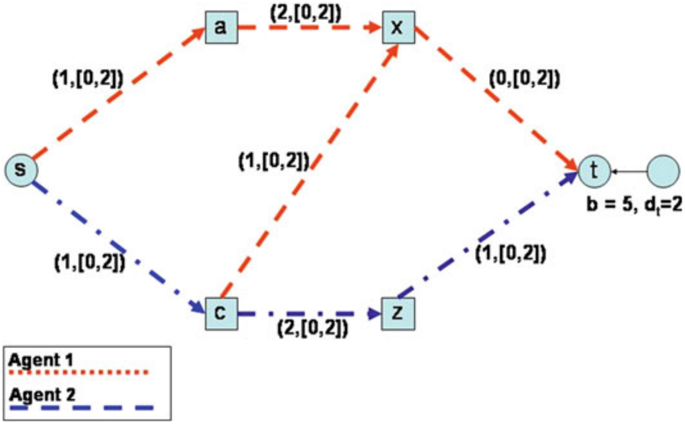


Fig. 16.6 Procurement graph with a non-empty core

the optimization problem given in Eqs. (16.1)–(16.5). The only difference between the two feasibility graphs is the valuation of the buyer for each unit of the item.

From an inspection of the characteristic function values in Table 16.3 corresponding to the feasibility graph in Fig. 16.5, it is clear that the core is non-empty. This is because there can be no allocation of the surplus that the coalition of both the agents can generate which is better than what each of the agents can generate on their own.

Now consider the procurement feasibility graph in Fig. 16.6 and the corresponding characteristic function values in Table 16.4. It is clear in this case that the feasibility graph admits an allocation that is in the core of the MPNF game.

Table 16.3 Characteristic function for the feasible procurement graph in Fig. 16.5

S	Surplus maximum flow F_S by S for $d_{v_t} = 2$	C_S	$v_{\mu_b}(S)$
{1}	2	6	6
{2}	2	8	4
{1, 2}	2	4	8

Table 16.4 Characteristic function for the feasible procurement graph in Fig. 16.6

S	Surplus maximum flow F_S by S for $d_{v_t} = 2$	C_S	$v_{\mu_b}(S)$
{1}	2	6	4
{2}	2	8	2
{1, 2}	2	4	6

From this discussion, we can conjecture that the core of the MPNF game may also depend on the valuation that the buyer attaches to each unit of the item.

16.3.3 Effect of the Maximum Demanded Quantity on the Core of the MPNF Game

We now try to develop intuition about the relationship between the maximum demanded quantity by the buyer and the non-emptiness of the core of the MPNF game. There are two things to check here:

- First is to see if we can transform a network with an empty core into one with a non-empty core by perturbing the maximum demand.
- Second, we need to check if we can transform a network with a non-empty core into one with an empty core by perturbing the maximum demand.

16.3.3.1 Effect of Perturbing Demand on a Feasibility Graph with an Empty Core

To study this effect, consider first the procurement feasibility graphs in Figs. 16.3 and 16.5 and the corresponding characteristic function values in Tables 16.1 and 16.3. Recall that the core of these feasibility graphs was empty.

We now perturb the demand upwards for both these graphs to $d_{v_t} = 3$ and $d_{v_t} = 4$, respectively. The new procurement feasibility graphs with the perturbed demands are shown in Figs. 16.7 and 16.8 and the corresponding characteristic functions in Tables 16.5 and 16.6.

From an inspection of the characteristic function values in Tables 16.5 and 16.6, we observe that the feasibility graphs now admit allocations that are in the core of the game. From these examples, we conjecture that by perturbing the

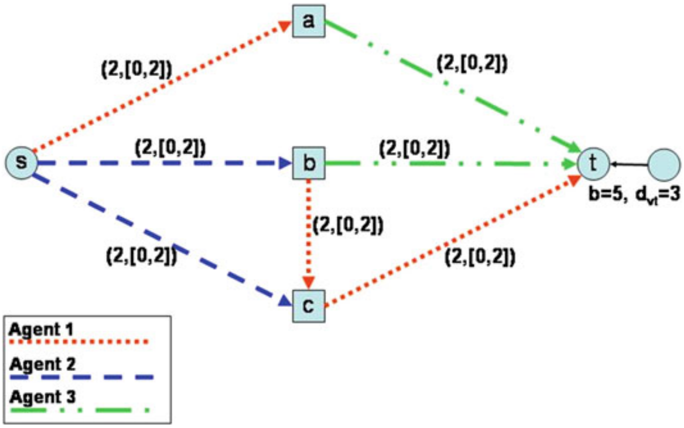


Fig. 16.7 Procurement graph admitting a non-empty

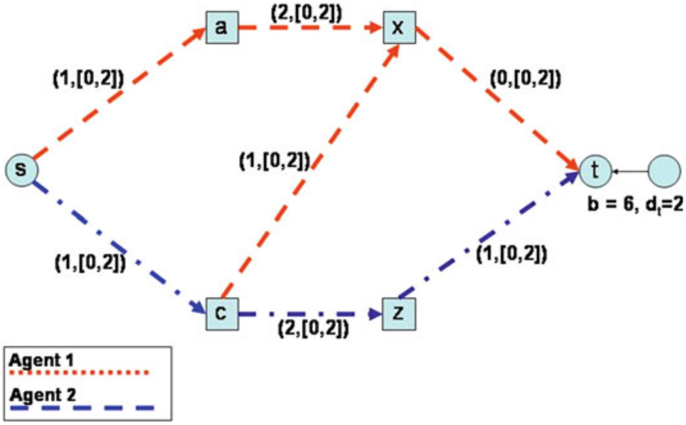


Fig. 16.8 Procurement graph admitting a non-empty core when demand is perturbed

Table 16.5 Characteristic function for the feasible procurement graph in Fig. 16.7

S	Surplus maximum flow F_S by S for $d_{v_t} = 2$	C_S	$v_{\mu_b}(S)$
{1}	0	–	0
{2}	0	–	0
{3}	0	–	0
{1, 2}	2	8	2
{2, 3}	2	8	2
{1, 3}	2	8	2
{1, 2, 3}	3	12	3

Table 16.6 Characteristic function for the feasible procurement graph in Fig. 16.8

S	Surplus maximum flow F_S by S for $d_{v_t} = 2$	C_S	$v_{\mu_b}(S)$
{1}	2	6	6
{2}	2	8	4
{1, 2}	4	14	10

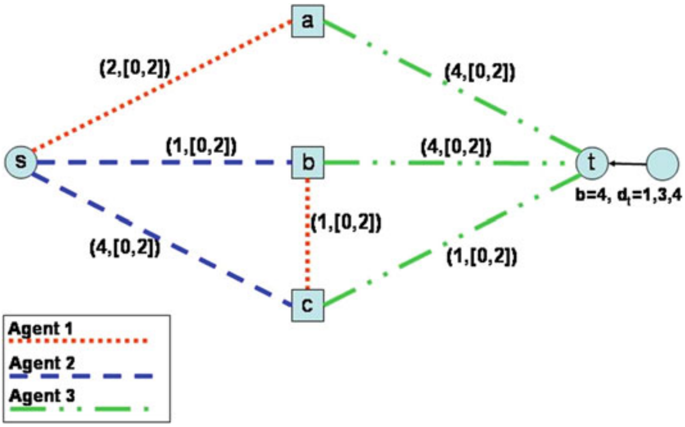


Fig. 16.9 Procurement graph with a non-empty core

maximum demand of networks which previously did not admit a core allocation, we can obtain a procurement feasibility graph with a non-empty core.

16.3.3.2 Effect of Perturbing Demand on a Feasibility Graph with a Non-empty Core

To study this effect, consider the procurement feasibility graphs in Figs. 16.4 and 16.6. Recall that these graphs have non-empty cores. We now perturb the demand. The procurement feasibility graphs with the perturbed demands are shown in Figs. 16.9 and 16.10. The characteristic function values for these graphs are given in Tables 16.7 and 16.8.

By an inspection of the characteristic functions for both these graphs, we observe that the demanded quantity does not seem to affect the non-emptiness of the core. Hence, it is difficult to make any precise conjecture of this action.

16.3.4 Summary of the Intuition

Through the examples in the previous sections, we have developed an intuition about the relationship between the non-emptiness of the core of the MPNF

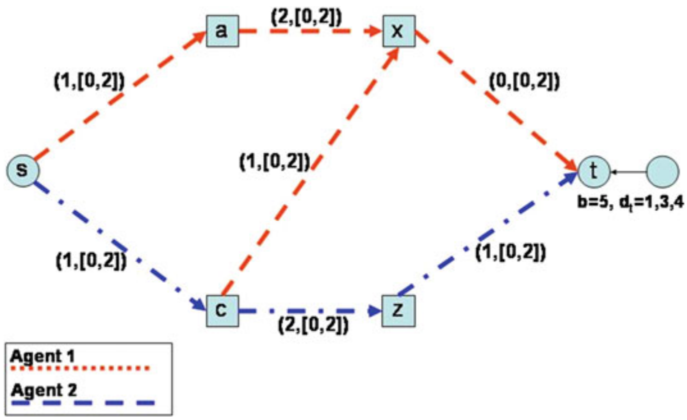


Fig. 16.10 Procurement graph with a non-empty core

Table 16.7 Characteristic function for the feasible procurement graph in Fig. 16.9

S	Surplus maximum flow $F_S$by S for $d_{v_t} = 1, 3, 4$	C_S	$v_{\mu_b}(S)$
{1}	0, 0, 0	–	0
{2}	0, 0, 0	–	0
{3}	0, 0, 0	–	0
{1, 2}	0, 0, 0	0	0
{2, 3}	0, 0, 0	0	0
{1, 3}	0, 0, 0	0	0
{1, 2, 3}	1, 2, 3	3, 6, 6	1, 2, 2

Table 16.8 Characteristic function for the feasible procurement graph in Fig. 16.9

S	Surplus maximum flow F_S by S for $d_{v_t} = 1, 3, 4$	C_S	$v_{\mu_b}(S)$
{1}	1, 2, 2	3, 6, 6	2, 4, 4
{2}	1, 2, 2	4, 8, 8	1, 2, 2
{1, 2}	1, 2, 4	2, 4, 10	3, 6, 6

problem $\mu(b) = (G, N, \psi, b, d_{v_t})$ and (a) the ownership structure $\psi(b)$ the valuation b of the buyer, and (c) the maximum demanded quantity d_{v_t} of the buyer. With the help of this intuition, we have conjectured the following:

- First, the structure of the network and the associated ownership structure seem to play an important part in determining non-emptiness of the core.
- Second, the buyer's valuation of the item, which in turn determines the surplus generated also seems to play a crucial role in determining the non-emptiness of the core.
- Finally, for any given procurement graph, there is a demanded quantity $d \in [0, \infty]$ such that for all $d_{v_t} \geq d$ the MPNF game generated by it seems to have a non-empty core.

We now examine these conjectures with an attempt to formalize the conditions under which non-emptiness of the core of the MPNF game can be ensured.

16.4 Conditions for Non-emptiness of the Core of MPNF Games

Before we move to examining formally the conditions for the non-emptiness of the core of the MPNF games, we first show that the MPNF game is a monotonic game.

Definition 16.1 A transferable utility (TU) game (N, v) is said to be monotonic if $v(S) \leq v(T)$ for all $S, T \subseteq N \; S \subseteq T$.

Proposition 16.1 *The characteristic function of the MPNF game is monotonically non-decreasing,* i.e., $v_{\mu b}(S) \leq v_{\mu b}(T), \forall S, T \subseteq N, S \subseteq T$

Proof We know that the characteristic function of the MPNF game is given by Eqs. (16.1)–(16.5).

$$v_{\mu b}(S) = \max \left[b x_{v_t} - \sum_{e \in E_S} c(e) f(e) \right] \tag{16.7}$$

subject to

$$\sum_{e \in I(j) \cap E_S} f(e) - \sum_{e \in O(j) \cap E_S} f(e) = 0, \quad \forall j \in N \setminus \{v_0, v_t\} \tag{16.8}$$

$$\sum_{e \in I(v_t) \cap E_S} f(e) = x_{v_t} \tag{16.9}$$

$$\sum_{e \in O(v_t) \cap E_S} f(e) = x_{v_t} \tag{16.10}$$

$$0 \leq x_{v_t} \leq d_{v_t}, \text{ and } l(e) \leq f(e) \leq u(e), \quad \forall e \in E_S \tag{16.11}$$

Let S, T be two coalitions of agents from N such that $S \subseteq T \subseteq N$. Clearly since the above characteristic function is derived from a maximization problem, a larger subset of agents means that there are no less number of edges through which flow can occur and hence there exist a larger set of options for maximizing the surplus. Therefore, $v_{\mu b}(S) \leq v_{\mu b}(T), \quad \forall S, T \subseteq N, S \subseteq T$ and the lemma holds.

In the rest of this section, we focus our attention on formalizing the conditions for the non-emptiness of the core of the MPNF game. To ease the discussion that follows, we introduce below some definitions.

Definition 16.2 Let $\mu(b) = (G, N, \psi, b, d_{v_t})$ be a MPNF situation where G is a directed acyclic graph whose edges are owned by agents in N as indicated by the ownership function $\psi : E \rightarrow N$, b is the buyer's valuation of a single unit of the item, and d_{v_t} is the maximum quantity demanded by the buyer. A flow F_S in G using only edges owned by agents in S is said to be a profitable flow in μb if $v_{\mu b}(N) > 0$.

Definition 16.3 Let μb be an MPNF situation as before. We associate with μb the MPNF game $(N, \mu b)$, then say that μb and $(N, \mu b)$ are non-trivial if μb has profitable flows or equivalently, if $v_{\mu b}(N) > 0$.

Definition 16.4 For any MPNF situation $\mu(b) = (G, N, \psi, b, d_{v_t})$ and its associated game $(N, \mu b)$, an agent $i \in N$ is called an f-veto agent (flow veto) if he owns at least one edge in every surplus maximizing flow in G.

16.4.1 Effect of Ownership Structure on the Non-emptiness of the Core

In this section, we focus on formalizing the intuition arising out of the discussion based on the procurement feasibility graphs in Figs. 16.3 and 16.4. There we saw that the ownership structure in the procurement feasibility graph played an important role in determining the non-emptiness of the core of the MPNF game. To formalize this relationship, we proceed by first showing that for any allocation of the surplus to be in the core of the MPNF game, it must be the case that positive allocations of surplus are only made to agents who are a part of the set of f-veto agents. With this result in hand, we then show that the set of f-veto agents must be non-empty and that these agents must own some critical edges in the feasibility graph if the core is to be non-empty.

Lemma 16.1 *Let $(N, v_{\mu b})$ be a non-trivial procurement network formation game with a set of f-veto agents v_f. Let x be an imputation of $(N, v_{\mu b})$ such that $x_i > 0$ for some agent $i \in N \backslash V_f$. Then $x \notin C(v_{\mu b})$.*

Proof Since $i \in N \backslash V_f$, $v_{\mu b}(N) = v_{\mu b}(N \backslash \{i\})$. Then, taking into account the fact that x is an imputation of $(N, v_{\mu b})$ and that $x_t > 0$,

$$v_{\mu b}(N \backslash \{i\}) = v_{\mu b}(N) = \sum_{j \in N} x_f > \sum_{j \in N \backslash \{t\}} x_f$$

That is, the surplus value allocated to agents in N\{i} is less than the surplus value that these agents can generate by themselves thereby violating the condition of the core. Hence $x \notin C(v_{\mu b})$.

Because of Lemma 16.1, we understand that the surplus is always divided among the members of the f-veto set v_f and a non f-veto agent never gets a positive share of the surplus. With this we can redefine the MPNF game to reflect the fact that only f-veto agents get a share of the surplus. That is the game $\left(N, v_{\mu b}\right)$ can be written as the game $\left(v_f, \hat{v}_{\mu b}\right)$ where the characteristic function $\hat{v}_{\mu b}(T)$ is the value that any subset $T \subseteq V_f$ can generate in conjunction with agents who are not in the f-veto set. That is,

$$\hat{v}_{\mu b}(T) = v_{\mu b}(T \cup (N \backslash V_f)) \tag{16.12}$$

With this definition of a new game involving only the agents in the f-veto set, we now state and prove the main theorem characterizing the effect of the edge ownership structure on the non-emptiness of the core of the MPNF game.

Theorem 16.1 *Let* $\left(N, v_{\mu b}\right)$ *be a non-trivial procurement network formation game with a set of f-veto agents* v_f. *Then,* $\left(N, v_{\mu b}\right)$ *is balanced if and only if the following two conditions hold*:

1. v_f *is non-empty and* $(v_f, \hat{v}_{\mu b})$ *is balanced.*
2. *Every profitable flow in the procurement network formation scenario* $\mu b = (G, N, \psi, b, d_{v_t})$ *with which* $\left(N, v_{\mu b}\right)$ *is associated contains an edge owned by an f-veto agent.*

Proof We first show the necessity part of the theorem. Suppose that $\left(N, v_{\mu b}\right)$ is balanced. We now need to show that conditions (1) and (2) of the theorem hold.

Now, from the theorem due to Bondereva (1963) and Shapley (1967), we know that the core of a transferable utility (TU) cooperative game is non-empty if and only if it is balanced. So, because of the assumption of balancedness of the MPNF game $\left(N, v_{\mu b}\right)$ which is a TU game, we can say that it has a non-empty core, i.e., $C\left(v_{\mu b}\right) \neq 0$.

Now, consider an imputation x of the surplus value such that it is in the core of the game $\left(N, v_{\mu b}\right)$. That is $x \in C\left(v_{\mu b}\right)$. Then from Lemma 16.1, v_f has to be a non-empty set and $x_i = 0$ for all $i \in N \backslash V_f$. Now, denote by $\hat{x}$ the restriction of x to the set v_f. Clearly, $\hat{x}$ is an imputation of $\left(v_f, \hat{v}_{\mu b}\right)$. Let $T \subset V_f$. Since $x \in C(v_{\mu b})$, we have:

$$\sum_{t \in T} \hat{x}_t = \sum_{t \in T \cup (N \backslash V_f)} x_t \geq v(T \ \cup \left(N \backslash V_f\right)) = \hat{v}(T).$$

Hence, $\hat{x} \in C\left(\hat{v}_{\mu b}\right)$, so $\left(v_f, \hat{v}_{\mu b}\right)$ is balanced and condition (1) of the theorem holds.

Now, suppose that there exists a profitable flow F such that $\psi(F) \cap V_f = \emptyset$. Then, $\sum_{t \in N \backslash V_f} x_t \geq v_{\mu b}(N \backslash V_f) > 0$. This contradicts Lemma 16.1 and hence there cannot exist a profitable flow where the flow is provided only by agents who are not a part of the set of f-veto agents v_f. Hence, condition (2) of the theorem holds. This proves the necessity part of the theorem.

We now show the sufficiency part of the theorem. Suppose that the MPNF game $(N, v_{\mu b})$ satisfies conditions 1 and 2. We now need to show that the core of this game is non-empty.

Consider an imputation $\hat{x} \in C(\hat{v}_{\mu b})$. We now define $x \in \Re^n$ in the following way: $x_i = \hat{x}_i$ if $i \in V_f$, else $x_i = 0$ for all $i \in N \backslash V_f$. It should be noted that the x we have constructed is now an imputation of $(N, v_{\mu b})$. Also we know that the game $(N, v_{\mu b})$ is monotonic from Proposition 16.1. Now consider any coalition of agents $S \subset N$. There are two cases to consider here: either (a) $S \cap V_f \neq \emptyset$ or (b) $S \cap V_f = \emptyset$.

Case (*a*): For every $S \subset N$ with $S \cap V_f \neq \emptyset$, since $\hat{x} \in C(\hat{v}_{\mu b})$, we have:

$$\sum_{i \in S} x_i = \sum_{i \in S \cap V_f} \hat{x}_i \geq \hat{v}_{\mu b}(S \cap V_f) \geq v_{\mu b}(S)$$

The above equation is nothing but the core condition for all coalitions $S \subset N$ where $S \cap V_f \neq \emptyset$.

Case (*b*): Now consider all coalitions $S \subset N$ where $S \cap V_f \neq \emptyset$. Since $(N, v_{\mu b})$ satisfies condition (2) in the statement of theorem, we can infer that for all coalitions S with $S \cap V_f \neq \emptyset$ we must have $v_{\mu b}(S) = 0$. It is easy to see that any allocation of the surplus of the MPNF game will always satisfy the core condition for all those coalitions S where $S \cap V_f \neq \emptyset$.

With this we have shown that for all coalitions $S \subset N$, we have $\sum_{i \in S} x_i \geq v_{\mu b}(S), \quad \forall S \subseteq N$. This is nothing but the condition for the non-emptiness of the core of the MPNF game $(N, v_{\mu b})$. So, sufficiency is proved. So the theorem holds.

16.4.1.1 Implications of Theorem 16.1 for Procurement Network Formation

The investigation of the relationship between the ownership structure and the non-emptiness of the core of the MPNF game has two implications: (i) it has design implications for procurement networks and (ii) it can also provide diagnostic support to identify potential sources of instability in the network.

First, from a procurement network design point of view, given the MPNF scenario $\mu b = (G, N, \psi, b, d_{v_t})$, the agents who have the maximum bargaining power in the feasibility graph can be marked out. Also, notice that the statement of the theorem says that the non-emptiness of the set of f-veto agents is the criterion for the core to be non-empty. Now assume that the buyer was himself included in the feasibility graph as an agent who owns a dummy edge from the terminal node v_t to another dummy node, say v_d and the feasibility graph now includes both the dummy edge and the node. It is easy to see that in this case, the set of f-veto agents is always non-empty and minimally includes the buying agent. By carrying out an analysis on this new graph, the buyer can see with which other suppliers in the network he will have to share the bargaining power. Such an analysis, often the subject of supplier footprint optimization within large global supply chains, can help the buyer to map out a strategy for supplier development so that the bargaining power of agents can be limited while simultaneously ensuring the stability of the network.

Second, from a diagnostic point of view, it provides support to the suppliers also. Recall that the result tells us that only f-veto agents get a positive share of the surplus. So, suppliers need to ensure that they are a part of the f-veto set if they hope to garner a share of the surplus. An analysis of the network will tell them how much they need to improve their cost competitiveness in order to be in the f-veto set. Also, it provides them with insight into what complementary capabilities or additional capabilities they need to acquire by buying out edges in the network that would make them a f-veto agent.

16.4.2 Effect of Buyer's Valuation on the Non-emptiness of the Core

In this section, we focus on formalizing the intuition arising out of the discussion based on the procurement feasibility graphs in Figs. 16.5 and 16.6. There we saw that the buyer's valuation of each unit of the item influences the characteristic function values of the MPNF scenario which in turn determines whether the core of the MPNF game is non-empty or not. We now formalize this intuition for a special case of the MPNF game where the maximum demanded quantity is unity.

To do this we proceed by first showing through Lemma 16.2 that if the special case of the MPNF scenario $\mu b = (G, N, \psi, b, d_{v_t} = 1)$ induces a game $(N, v_{\mu b})$ whose core is non-empty, then for every other MPNF scenario where the buyer's valuation b of the item is lower, the induced game $(N, v_{\mu b})$ continues to have a non-empty core. We then show through Theorem 16.2 that either the core of the MPNF game is always non-empty or there exists a threshold value of the buyer's valuation below which the core is always non-empty.

Lemma 16.2 *Let $\mu b = (G, N, \psi, b, d_{v_t} = 1)$ be a procurement network formation scenario and the associated cooperative game be $(N, v_{\mu b})$. If $(N, v_{\mu b})$ is balanced, then $(N, v_{\mu b})$ is balanced for all $\hat{b} \in [0, b]$ where $\mu\hat{b} = (G, N, \psi, \hat{b}, d_{v_t} = 1)$.*

Proof Let $x = (x_1, x_2, \ldots, x_n)$ be an imputation of $(N, v_{\mu b})$. Then x is an element of the core of the game if and only if $\sum_{i \in S} x_i \geq v_{\mu b}(S)$ for every $S \subseteq N$ that has a profitable flow in μb. Now, consider some $x = (x_1, x_2, \ldots, x_n) \in C(N, v_{\mu b})$ and any $\hat{b} \leq b$. The corresponding procurement network formation scenario μb induces the cooperative game $(N, v_{\mu b})$. There are two cases to consider for the newly induced game. Either (a) $v_{\mu b}(N) = 0$ or (b) $v_{\mu b}(N) > 0$

Case (a): When $v_{\mu b}(N) = 0$.

If $v_{\mu b}(N) = 0$, then because the MPNF game is monotonic, it is clearly the case that there is no subset of agents who can create a positive surplus. That is $v_{\mu b}(S) = 0, \ \forall S \subseteq N$. Then for any balanced vector $\theta = (\theta(S))_{S \in L(N),}$ where $L(N) = \{S | S \subseteq N, S \neq \emptyset\}$, it is clear that $\sum_{S \subseteq N} \theta(S)\ v_{\mu\delta}(S) = 0 = v_{\mu\delta}(N) = 0$. This is nothing but the condition for balancedness of the game $(N, v_{\mu b})$ (refer the Theorem 16.1). Hence the lemma holds in this case.

Case (b): When $v_{\mu b}(N) > 0$.

For this note that the difference in surplus that the grand coalition makes when the buyer's valuation is b and $\hat{b}$ is exactly equal to the difference in the valuation itself. This is given by the following equation:

$$v_{\mu b}(N) - v_{\mu\delta}(N) = b - \hat{b} \geq 0 \tag{16.13}$$

We have already picked $x = (x_i)_{i \in N}$ as an imputation of the surplus that is in the core of the game $(N, v_{\mu b})$. Now we choose an imputation $\hat{x}$ of $(N, v_{\mu\delta})$ such that the following condition holds:

$$x_i - \hat{x} \geq 0, \quad \forall i \in N \tag{16.14}$$

From Eqs. 16.13 and 16.14, we can deduce the following relation:

$$\sum_{i \in S} x_i - \sum_{i \in S} \hat{x}_i \leq b - \hat{b} \tag{16.15}$$

$$\begin{aligned}
&\Rightarrow \textstyle\sum_{i \in S} \hat{x}_i \geq \sum_{i \in S} x_i + \hat{b} - b \\
&\Rightarrow \textstyle\sum_{i \in S} \hat{x}_i \geq \sum_{i \in S} x_i + b + \hat{b} \\
&\Rightarrow \textstyle\sum_{i \in S} \hat{x}_i \geq b - c(F_s) - b + \hat{b} \\
&\Rightarrow \textstyle\sum_{i \in S} \hat{x}_i \geq \hat{b} - c(F_s) \\
&\Rightarrow \textstyle\sum_{i \in S} \hat{x}_i \geq v_{\mu\delta}(S)
\end{aligned}$$

The first and second inequalities following inequality 16.15 is simply a rearrangement of terms; the third inequality follows from the core condition for the original game $(N, v_{\mu b})$ where $\sum_{i \in S} x_i \geq v_{\mu b}(S) \geq b - c(F_s)$, and the last inequality follows from the rearrangement of terms in the previous inequality and substitution for the core condition. So starting from an imputation which is in the core of the original game, we have shown that it is possible to construct an imputation which is in the core of the new game whose budget is bounded by the budget of the original game for which the core is non-empty. And by Theorem due to Bondereva (1963), we know that the core of a cooperative game is non-empty if and only if it is balanced. Since the new game has a non-empty core, we can infer that it is also balanced and the lemma holds for this case also.

So, the Lemma holds.

We can now state and prove the main theorem characterizing the effect of the buyer's valuation on the non-emptiness of the core of the MPNF game where the maximum demanded quantity by the buyer is unity.

Theorem 16.2 *Let $\mu b = (G, N, \psi, b, d_{v_t} = 1)$ be a procurement network formation scenario. Then either the cooperative game $\left(N, v_{\mu b}\right)$ associated with the scenario is balanced for all $b \in [0, \infty]$ or there exists $B \in [0, \infty]$ such that $(N, v_{\mu b})$ is balanced if and only if $b \leq B$.*

Proof Suppose that there exists $\tilde{b} \in [0, \infty]$ such that$(N, v_{\mu\tilde{b}})$ is not balanced. Now consider the set of real numbers given by:

$$R = \{b \in (0, +\infty) | \quad C(v_{\mu d}) \neq \emptyset, \quad \forall\, d \in [0, b]\} \tag{16.16}$$

Clearly $R \neq \emptyset$, because $0 \in R$. Also, $\tilde{b}$ is an upper bound of R. Hence the supremum $B \in [0, \infty]$ of R should exist.

Now, because of Lemma 16.2, to prove the theorem, we only need to prove that $B \in R$. Equivalently, we need to show that the core of the MPNF game $(N, v_{\mu B})$ with valuation B is non-empty, i.e., $C(v_{\mu B}) \neq \emptyset$

If we are able to show this, then because of Lemma 16.2, we can make the claim that for every $b \leq B$, the core of the game $\left(N, v_{\mu b}\right)$ is non-empty. We now proceed to show this.

Now consider for each $t \in \mathbb{N}, b^t \in R$ such that $B - b^t \leq 1/t$ and $x^t \in C\left(v_{\mu b^t}\right)$. Under these conditions, it is clear that the sequence of surplus allocation $\{x^t\}_{i \in N}$, where $x^t = (x_i^t)_{i \in N}$, has a convergent subsequence. Without loss of generality, we can identify this subsequence with $\{x^t\}_{i \in N}$. We denote the limit of $\{x^t\}_{i \in N}$ by $\bar{x}$.

Now, for all $t \in (N)$ we have chosen $x^t \in C\left(\left(N, v_{\mu_b^t}\right)\right)$. This means $x_i^t \geq v_{\mu_{b^t}}(i)$ and $\sum_{i \in N} x_i^t = v_{\mu_{b^t}}(N)$. Because of this, we have $\bar{x}_i \geq v_{\mu B}(i)$ and $\sum_{i \in N} \bar{x}_i = v_{\mu B}(N)$. Hence $\bar{x}$ is an imputation of $(N, v_{\mu_{b^t}})$.

Now, for each $t \in (N)$, since we have chosen $x^t \in C(v_{\mu bt})$, for every $S \subset N$ we have $\sum_{i \in S} x_i^t = v_{\mu_{b^t}}(S)$. Taking limits as t goes to infinity, we obtain

$$\sum_{i \in S} \bar{x}_i \geq v_{\mu B}(S), \quad \forall\, S \subset N$$

The above inequality is nothing but the condition for the non-emptiness of the core. So, we have $\bar{x} \in C(v_{\mu B})$.

This means that we have on hand a valuation $B \in [0, \infty]$ such that the induced game $\left(N, v_{\mu B}\right)$ has a non-empty core. And from Lemma 2 we know that for every $b < B$, the induced game $\left(N, v_{\mu B}\right)$ has a non-empty core and hence the theorem holds.

16.4.2.1 Implications of Theorem 16.2 for Procurement Network Formation

From a managerial perspective, Theorem 16.2 suggests that if the buyer desires the formation of a stable procurement network in the sense that recontracting is precluded, then the budget that he announces will have to be carefully chosen. A very low budget may mean that there is no profitable flow in the network and hence a transaction does not occur and a high budget will tend to engage the suppliers in protracted negotiations which do not seem to terminate in any agreement precisely because the core is empty.

16.4.3 Effect of Demanded Quantity on the Non-emptiness of the Set of f-veto Agents

In this section, we focus on formalizing the intuition arising out of the discussion based on the procurement feasibility graphs that we considered in Figs. 16.7 and 16.8 and in Figs. 16.9 and 16.10. We observed there that given a feasibility graph, the demanded quantity seemed to play a part in determining the non-emptiness of the core. From Lemma 16.1 and Theorem 16.1, we understand that the existence of a non-empty set of f-veto agents is crucial to the non-emptiness of the core of the MPNF game. In this section, we formally show the relationship between the demanded quantity and the non-emptiness of the core through its relationship with the non-emptiness of the set of f-veto agents. We capture this relationship in Theorem 16.3.

Theorem 16.3 *Let* $\mu b = (G, N, \psi, b, d_{v_t})$ *be a* MPNF *scenario. Then there exists* $D \in (0, \infty)$*such that the set of f-veto agents is non-empty for all* $d_{v_t} \geq D$.

Proof Recall that the set of f-veto agents V_f are those agents that own an edge in every surplus maximizing flow in the network. Because of this, we can infer that there are at least two coalitions of agents $S_1, S_2 \subseteq N, S_1 \cap S_2 = \emptyset$ who can provide a surplus maximizing flow for a given procurement scenario $\mu b = (G, N, \psi, b, d_{v_t})$ when the set V_f is empty. That is we have:

$$v_{\mu b}(N) = v_{\mu b}(S_1) = v_{\mu b}(S_2) \quad (16.17)$$

It is clear that the limiting constraint in this surplus maximization problem is the maximum demanded quantity d_{v_t}. Set $d_{v_t} \Leftarrow 2 * d_{v_t}$. Now, $S_1 \cup S_2$ is a surplus maximizing coalition for the procurement scenario $\mu b = (G, N, \psi, b, 2 * d_{v_t})$. Now, either $S_1 \cup S_2$ yields a f-veto set of agents in which case we are done or we repeat the procedure until such a set is found. And this is guaranteed since there are only a finite set of agents. So, by this procedure we are guaranteed to find a number $D \in (0, \infty)$ such that $d_{v_t} = D$ induces a set of f-veto agents.

16.4.3.1 Implication of Theorem 16.3 for Procurement Network Formation

Recall from Theorem 16.1 that one of the conditions for the core to be non-empty is that the set of f-veto agents should be non-empty. Theorem 16.3 is essentially a link to the non-emptiness of the core through the set of f-veto agents. From a managerial perspective, the theorem suggests that if the buyer desires the formation of a stable procurement network, then the maximum demand that he specifies will have to be carefully chosen if the set of f-veto agents is to be non-empty.

16.5 An Extensive-Form Game to Implement the Core of the MPNF Game

In general, the core as a solution concept only points out that an allocation in the core is immune to recontracting either by the grand coalition of all agents or sub-coalitions of agents. It takes an exogenous view of the cooperative scenario and points out that agents when given sufficient time and message space for negotiations are likely to converge to one of the allocations in the core that are themselves indicated axiomatically. From an implementation viewpoint, however, this can be cumbersome. That is, in actual practice, in a MPNF scenario, we do not normally have the advantage of a central agency or a social planner who can point out to the agents the strategies they must adopt to obtain payoffs that are in the core of the game. We must have a way to allow agents to non-cooperatively achieve the desired outcome.

As we know the core can essentially be viewed as a social choice correspondence from the space of characteristic function values to the space of allocations. That is, the core C of a cooperative game (N, v) is a social choice correspondence given by:

$$C : \Re^{|L(N)|} \to \Re^{N} \quad \text{where } L(N) = \{S | S \subseteq N,\ S \neq \emptyset\}$$

Implementation theory provides us with a body of ideas to implement such social choice correspondences.

We point the reader to Maskin (1985) and Moore (1992) for a comprehensive survey of this field. The idea here is to construct detailed rules of bargaining in an extensive-form game and show that the set of non-cooperative equilibria of this game, possibly Nash equilibria or some refinement thereof, coincide with the outcomes of the social choice correspondence. In this section, we draw from these ideas, and in particular from the mechanism of Serrano and Vohra (1997) to design a non-cooperative game such that the outcomes corresponding to the equilibrium strategies of each of the agents of such a game correspond to the outcomes in the core of the MPNF game whenever it is non-empty.

16.5.1 Preliminaries

We shall now construct an extensive-form game such that for every scenario in the class of MPNF games, the sub-game perfect Nash equilibrium outcomes coincide with the core allocations. To do that, we first need some additional notation regarding (a) the outcomes that are achievable by any sub-coalition of agents and (b) the notion of permutations and their composition.

For this part of the discussion, we treat each of the edges in the network as individual resources that are owned by agents. Since there are |E| edges, we designate each agent i's initial endowment of resources as $r_i^0 \in [0,1]^{|E|}$ and the final allocation of resources after the formation of the procurement network as $r_i \in [0,1]^{|E|}$.

We know from the earlier discussion in this chapter that the surplus achievable by coalition S is given by $v_{\mu b}(S)$. If S can provide a profitable flow through the network then $v_{\mu b}(S) > 0$. We let $(x_i)_{i \in S}$ be an allocation of the surplus such that $\sum_{i \in S} x_i = v_{\mu b}(S)$. We now define an outcome achievable by a coalition S as a vector $(r_i, x_i)_{i \in S}$ that indicates a reassignment of resources that follows from solving the optimization problem specified in Eqs. (16.1)–(16.5) and an allocation of the surplus $v_{\mu b}(S)$ if the coalition is able to find a profitable flow through the network using only the edges owned by the agents in the coalition S. We define Ω_S as the set of outcomes achievable by the coalition $S \subseteq N$ and let $\Omega = \cup_{S \subseteq N} \Omega_S$. Formally, we have:

$$\Omega_S = \left\{ (r_i, x_i)_{i \in S} | \sum_{i \in S} r_i \leq \sum_{i \in S} r_i^0, \text{ and } \sum_{i \in S} x_i \leq v_{\mu b}(S) \right\} \tag{16.18}$$

An important idea, first introduced by Thomson in (Thomson 2005) to implement solutions to problems of fair division, that we use to define the extensive-form game, to implement the core of the MPNF game is the notion of permutations. We let $\prod$ denote the set of all permutations of N, i.e., one-to-one functions from N to N. Given $\pi = (\pi_i)_{i \in N}$ where $\pi_i \in \prod$ for every $i \in N$, we define $p(\pi)$ to be the composition of all permutations (π_i).

That is $p(\pi) = \pi_1(\pi_2(\ldots(\pi_i(\ldots\pi_N)\ldots)))$ and $p(\pi)_j$ is the jth agent in the sequence $p(\pi)$. Notice that from the way in which the composition of permutations is defined, given any $\pi_{-i} = (\pi_1, \pi_2, \ldots, \pi_{i-1}, \pi_{i+1}, \ldots, \pi_N)$ and $\hat{\pi} \in \Pi, \pi_i \in \Pi$ there exists $\bar{\pi} \in \Pi$ such that $p(\bar{\pi}, \pi_{-i}) = \hat{\pi}$. This means that any agent $i \in N$ can make a unilateral change in $\pi_i \in \prod$ so that when the composition is computed he ends up as the first agent in the computed order.

16.5.2 Description of the Extensive-Form Game

The extensive-form game or mechanism that we construct is essentially a two stage game with agents moving simultaneously in Stage 0 and moving sequentially in Stage 1. Our aim for Stage 0 is twofold: First, we would like to have a status quo allocation that agents will revert to in case of disagreement in Stage 1; second we seek an ordering of agents so that they may move sequentially in Stage 1. In Stage 1, we allow agents to make sequential moves, such as proposing, accepting, or rejecting offers. These moves are based on an ordering of agents computed at the end of Stage 0. The game ends after each of the agents has made at most one move in Stage 1. The leaf nodes of the game tree indicate whether an agreement has been reached and if so a split of the surplus. We now present the details of the extensive-form game Γ to implement the core of the MPNF game.

To construct the extensive-form game, we follow the game structure provided in Mas-Collel et al. (1995). The extensive-form game Γ to implement the core of the MPNF game $(N, v_{\mu b})$consists of the following building blocks:

1. A finite set of agents *N*, a finite set of nodes *N*, and a finite set of actions A.
2. The set of agents $N = \{1, 2, \ldots, n\}$ corresponds to the set of agents who own edges in the MPNF scenario $\mu b = (G, N, \psi, b, d_{v_t})$
3. A function *pr*: $N \rightarrow \{N \cup \emptyset\}$ specifying a single immediate predecessor *pr*(*k*) of each node$k \in N$. It is non-empty for all $k \in N$ except for one node which is designated as the initial node k^0. The immediate successor nodes of *k* are then $su(k) = pr^{-1}(k)$ and the set of all predecessors and successors can be found by iterating *pr*(*k*) and *su*(*k*). To have a tree structure, we require that these sets be disjoint, i.e., a predecessor of a node cannot also be a successor of the node. The set of leaf nodes of the tree is $L = \{k \in N | su(k) = \emptyset\}$. All other nodes *N\L* are called decision nodes. These decision nodes may be divided into two sets N^0 and N^1 which contain the decision nodes in Stage 0 and Stage 1 of the game, respectively.
4. The action set *A* consists of two sets A^0 for Stage 0 and A^1 for Stage 1. So, $A = A^0 \cup A^1$.

- In Stage 0, every agent chooses simultaneously an action $a_i^0 = (\omega_i, \pi_i)$ from the action set $A^0 = \{(\omega_i, \pi_i) | \omega_i \in \Omega; \pi_i \in \Pi\}$. Here, ω_i is an indication of an outcome that agent *i* would like to see emerge as the status quo outcome at the end of Stage 0. The outcome $\omega_i \in \Omega$ is a tuple $(r_i^j, x_i^j)_{j \in N}$ that specifies a reassignment of the edges $((r_i^j)_{j \in N})$ and an allocation of surplus $((x_i^j)_{j \in N})$ among the members of the coalition *N*. In other words, $((x_i^j)_{j \in N})$ indicates how agent *i* would like to split the surplus among the set of all participating agents in the MPNF game. The second part of the action a_i^0 specifies a choice of permutation $\pi_i \in \Pi$ that *agent i* would like to have implemented in the next stage of the game. Let $a^o = (a_i^0)_{i \in N}$ be the profile of Stage 0 messages where

$a_i^0 \in A^0$. Now denote $1(a^0) = p(\pi)_1, 2(a^0) = p(\pi)_2, \ldots, n(a^0) = p(\pi)_n$ as the first, second,... and the last agent as decided by the order $p(\pi)$. If for any i and j such that $i \neq j$, we have $\omega_i \neq \omega_j$ then we penalize agent $N(a^0)$ by an amount ε_i. For all practical purposes, we may treat this as a penalty levied on some caution deposit that the agents make before the start of the game. If all things go well, then this caution deposit is given back to the agents along with any of the surplus that they may be entitled to at the end of the game. If, however, $\omega_i = \omega_j = \omega^*$for all $i,j \in N$, then ω^* we designate as the status quo allocation and move to the next stage of the game.

- In Stage 1, agents choose sequentially an action a_i^1 from the action set A^1. The sequence of agents is determined by the composition $p(\pi)$ of all the permutations $\pi = (\pi_i)_{i \in N}$ submitted by agents in Stage 0. The action that an agent i can take in this stage is one of the following:

– Make an offer (S, ω_i) to coalition S to implement an outcome $\omega_i \in \Omega_S$. The outcome ω_i is a tuple $(f_i^j)_{j \in S}$ indicating a reassignment of the edges $(f_i^j, x_i^j)_{j \in S}$ so that the procurement network is formed and a division $(x_i^j)_{j \in S}$ of the surplus $v_{\mu b}(S)$.
– Accept an offer (S, ω_i) made by a preceding agent i in stage 1.
– Reject an offer (S, ω_i) made by a preceding agent i in stage 1.
So, the elements of the action set A^1 are as follows:
$A^1 = \{$make offer, accept offer, reject offer $(S, \omega_i) | S \subset N$ and $\omega_i \in \Omega_S\}$.
The first agent $1(a^0)$ in the sequence determined by $p(\pi)$makes an offer$(S, \omega_i) \in A^1$. The other members of S then respond sequentially in the order $2(a^0), \ldots, k(a^0)$ where $j(a^0)$ is the jth agent in S as determined by the sequence$p(\pi)$. They either accept it or reject it. If all members of S accept the offer, then they receive the allocations $(x_i^j)_{j \in S}$ as offered by agent $i = 1(a^0)$ and all other agents not in S receive no allocations of the surplus. If even one of the members of S rejects the offer, then the status quo offer from Stage 0 is implemented.

5. A function $\alpha : N \backslash \{k_0\} \rightarrow$A gives the action that leads to any non-initial node k from its immediate predecessor node pr(k) and satisfies the property that if $\hat{k} \neq \bar{k} \in su(k)$and $\hat{k} \neq \bar{k}$, then $\alpha(\hat{k}) \neq \alpha(\bar{k})$. The set of actions available at decision node k is $a(k) = \{a \in A : a = \alpha(k),$ for some $\bar{k} \in \text{su}(k)\}$.
6. A collection of information sets H, and a function $H : N \rightarrow H$assigning each decision node to an information set $H(k) \in H$. Thus the information sets H make a partition of the set of nodes N. We require that all decision nodes assigned to a single information set have the same actions available at every one of those nodes. That is, $a(k) = a(\hat{k})$ if $H(k) = H(\hat{k})$. We can therefore write the actions available at information set H as $A(H) = \{a \in A : a \in a(k)$ for $k \in H\}$.
7. A function $I : H \rightarrow N$, assigning each information set in H to the agent who moves at the decision nodes in that set. We can denote the collection of agent i's information sets by $H_i = \{H \in H : i = I(H)\}$.

8. A strategy for an agent i is a function $s_i : H_i \to A$ such that $s_i(H) \in A(H)$ for all $H \in H_i$. We now denote

- the set of all possible strategies for an agent i by S_{i}.
- the strategy profile for the coalition of all agents in N as $s = (s_1, s_2, \ldots, s_{\mathrm{n}})$.
- the set of all possible strategy profiles of all agents in N as $S = X_{i \in N} S_i$.

9. A function $g : S \times N \to \Omega$ where S, N, and Ω are the set of strategy profiles, set of nodes in the game tree, and the set of outcomes, respectively. Note that each of the leaf nodes L corresponds to an outcome from the set. So, for a strategy profile $s \in S$, we let $g(s, k)$ denote the outcome corresponding to s starting at node k.
10. Finally, we define a collection of payoff functions $u = \{u_1(\cdot), u_2(\cdot), \ldots, u_n(\cdot)\}$ that agents have for each of the outcomes. That is $u_i : \Omega \to \Re$ and $u : \Omega \to \Re^n$. We know that an outcome $\omega \in \Omega$ is given by the tuple $\omega = (r_i, x_i)_{i \in N}$. We let the utility that agent i has for an outcome $\omega \in \Omega$be given by $u_i(\omega) = x_i$.

So, for the MPNF game in characteristic function form $(N, v_{\mu b}())$, we have specified the extensive-form game Γ by the tuple:

$$\hat{x}_i \neq x_i \; \Gamma = (N, A, N, \mathrm{pr}(.), \alpha(.), H, H(.), I(.), u) \tag{16.19}$$

We know that a profile of strategies $s = (s_1, s_2, \ldots, s_n)$ in an n-player extensive-form game is a sub-game perfect Nash equilibrium (SPNE) if it induces a Nash equilibrium in every sub-game. So, in the extensive-form game Γ that we have defined above for the MPNF game in characteristic function form, the SPNE of Γ is a strategy profile $s^* \in S$ such that for all $k \in N \backslash L$ and for all $i \in N$, we have:

$$u_i\big(g\big(s_i^*, s_{-i}^*, k\big)\big) \geq u_i\big(g\big(s_i^*, s_{-i}^*, k\big)\big), \quad \forall s_i \in S_i \tag{16.20}$$

We let SPNE (Γ) denote the set of all allocations corresponding to the sub-game perfect Nash equilibria of the extensive-form game Γ. An extensive-form game Γ is said to implement in sub-game perfect Nash equilibrium the core of a MPNF game if we have the following (with some abuse of notation):

$$SPNE(\Gamma) = C((N, v_{\mu b})) \tag{16.21}$$

16.5.3 Analysis of the Extensive-Form Game

The analysis of the mechanism essentially involves verifying if the extensive-form game Γ constructed above makes the agents in the game pick equilibrium strategies whose payoffs correspond to the core allocations of the corresponding game specified in its characteristic function form. We do this by proving Theorem 16.4 below.

Theorem 16.4 *The extensive form mechanism Γ implements in sub-game perfect Nash equilibrium the core of the class of MPNF games.*

Proof To prove this theorem, we essentially need to show two things:

a. First, we need to show that any outcome, and hence an allocation of the surplus, that is in the core of the MPNF game corresponds to the outcome of a strategy profile that is an SPNE of the extensive-form game Γ.
b. Second, that an outcome corresponding to a strategy profile that is an SPNE of the extensive-form game Γ corresponds to an outcome, and hence an allocation of the surplus, that is in the core of the MPNF game.

Part (a): We proceed by first showing that if there is a surplus allocation vector $x \in C((N, v_{\mu b}))$ then there is an outcome $\omega \in \mathrm{SPNE}(\Gamma)$ such that the payoff $u = (u_i)_{i \in N}$ corresponding to the outcome ω is equal to x.

Consider a strategy profile $s = (s_1, s_2, \ldots, s_n)$ such that:

i. In Stage 0, every agent i chooses an action $s_i^0 = (\omega, \pi^0)$ where the outcome ω is chosen such that $u(\omega) = x$ and π^0 is the identity permutation.
ii. In Stage 1, every agent *i* chooses, at every node that he has to make an offer, an action $(S, \omega_i) = (N, \omega)$. The outcome ω is chosen such that $u(\omega) = x$.
iii. At every node of Stage 1 where an agent has to respond to a status quo outcome $\tilde{\omega}$ and an offer $(S, \hat{\omega})$, he accepts the offer only if $u_i(\hat{\omega}) > u_i(\tilde{\omega})$.

Consider any decision node in Stage 1 of the game where an agent has to respond to a proposal vis-a-vis a status quo outcome. If his action is such that it satisfies condition (iii) above, it is clear that it corresponds to a sub-game perfect Nash equilibrium starting at any of the nodes at which the agent has to respond to an offer.

Because, the payoff x corresponding to the status quo outcome ω is in the core of the MPNF game, no agent can object to the outcome ω and propose an alternative outcome at any node in Stage 1 where the agent has to make a proposal vis-a-vis the status quo outcome. This implies that actions taken in accordance to condition (ii) above correspond to a sub-game perfect Nash equilibrium action for any agent starting at any node where he has to make a proposal vis-a-vis the status quo outcome.

Now, since the payoff x corresponding to the outcome ωs in the core of the game, no agent i can gain by choosing an outcome $\hat{\omega} \neq \omega$ which gives him a payoff $\hat{x}_i \neq x_i$. This means that the strategy profile $s = (s_1, s_2, \ldots, s_n)$ chosen as above leads us to an equilibrium path where

- the status quo at the end of Stage 0 is the outcome ωwhose payoff is $u(\omega) = x$,
- the first agent in the sequence computed at the end of Stage 0 proposes an alternative that includes the grand coalition and the outcomeω,
- and all other agents reject it in favor of the status quo outcome at all nodes in Stage 1 where they have to respond to a proposal made in Stage 1.

This means that if we are given any allocation that is in the core of the MPNF game, we can always find in the extensive-form game Γ a sub-game perfect Nash equilibrium strategy profile $s = (s_1, s_2, \ldots, s_n)$ whose outcome gives the same payoff. This proves Part (a).

Part (b): We now need to show that if Γ has a sub-game perfect Nash equilibrium strategy profile $s = (s_1, s_2, \ldots, s_n)$, then its outcome $g(s, k^0)$ has a utility profile $u(g(s, k^0))$ that corresponds to an allocation in the core of the MPNF game, i.e., $u(g(\bar{s}, k^0)) = x \in C((N, v_{\mu b}))$.The proof is by contradiction. But before that we need to establish Lemmas (16.3)–(16.5) below.

Consider a strategy profile $s = (s_1, s_2, \ldots, s_n)$ that is a sub-game perfect Nash equilibrium of the game Γ. The actions $a_i^0 = (\omega_i^0, \pi_i)$ that each of the agents i take in Stage 0 of the game correspond to a status quo outcome whose allocation of payoffs is $(x_i^0)_{i \in N}$. And we also let $g(s, k^0) = \omega$ whose payoffs correspond to $u(g(s, k^0)) = u(\omega) = (x_i)_{i \in N}$.

Lemma 16.3 *For the given sub-game perfect Nash equilibrium strategy profile s, the outcomes and hence the resulting payoff allocations announced by each of the agents in Stage* 0 *are the same. That is,* $\omega_i^0 = \omega^0$ *for all* $i \in N$ *so that* $x_i^0 = x^0$.

We prove this lemma by contradiction. Suppose that the claim is not true. Then according to the rules of the extensive-form game, agent j who is the last in the sequence computed by $p(\pi)$ suffers a penalty ε_j. However, this agent can easily avoid this penalty by changing his action $a_j^0 = (\omega_j^0, \pi_j)$ to $\hat{a}_j^o = (\omega_j^0, \hat{\pi}_j)$ so that $j \neq p(\hat{\pi}_j, \pi_{-j})_n$, i.e., agent j is not the last in the computed sequence.

This means that the agent j stands to gain by changing his action in Stage 0 from a_j^0 to $\hat{a}_j^o$ which contradicts the hypothesis that the strategy profile $s = (s_1, s_2, \ldots, s_n)$ where the action a_j^0 chosen in Stage 0 by agent j is a SPNE. So, the claim holds.

Lemma 16.4 *The utility that each agent has for the final outcome is at least as great as the utility that they receive from the status-quo outcome. That is,* $u(\omega) \geq u(\omega^0)$ *implying* $x \geq x^0$.

We prove this lemma again by contradiction. Suppose that there is an agent i who gets less utility/payoff from the final outcome as compared to the status quo outcome. Then, in Stage 0 of the game, the agent can change his announcement of the permutation π_i to $\hat{\pi}_i$ so that he becomes the first agent in the sequence computed at the end of Stage 0. We know that this is possible from our earlier discussion on the notion of permutations. Now that he is the first agent in the sequence computed, he can propose the status quo outcome and irrespective of how other agents respond, the status quo outcome can be ensured as the final outcome. This is a contradiction since we started off with the assumption that S is an SPNE strategy profile. Hence, the claim holds.

Lemma 16.5 *Let k be any node of any sub-game in which a player has to respond to the status-quo outcome* ω^0 *having a payoff* x^0 *and a Stage* 1 *offer* $(S, \hat{\omega})$ *having a* *payoff* $\hat{x}$ *made by agent i. Suppose that* $u_j(\hat{\omega}) > u_j(\omega^0)$ *for all* $j \neq i, j \in S$. *Then* $g(s,k) = \hat{\omega}$ *in the extensive form game* Γ.

This is a straight forward consequence of the sub-game perfectness of the strategy s. An agent who is responding to a proposal must accept it if it gives him more utility or payoff when compared to the utility that he gets from a status quo outcome.

Now, to complete the proof, assume that the payoff corresponding to the outcome of the SPNE strategy profile is not in the core of the MPNF game, i.e., we assume that $u(g(s,k^0)) = (x_i)_{i \in N} \notin C(v_{\mu b})$.

Because the payoff is not in the core of the MPNF, there will be a coalition of agents S who can find an outcome ω_S whose payoff allocation $(\bar{y}_j)_{j \in S}$ is better than what they would receive from the outcome of the SPNE strategy profile s and hence there would be an objection (S, ω_S). That is, for all agents $j \in S$ we have $(\bar{y}_j)_{j \in S} \geq (x_j)_{j \in S}$. From claim 2, we know that $x \geq x^0$. So, we have:

$$(\bar{y}_j)_{j \in S} \geq (x_j)_{j \in S} \geq (x_j^0)_{j \in S} \tag{16.22}$$

This immediately implies that any agent $j \in S$ can unilaterally change his announcement of the permutation from π_j to $\hat{\pi}_j$ when he chooses his action in Stage 0 of the game so that he becomes the first agent in the sequence for stage 1. He can then choose an action which proposes (S, ω_S) which is sure to be accepted by all agents in S. But this would mean that the original strategy profile is not in equilibrium which is a contradiction. Hence, the payoff that corresponds to a strategy profile which is SPNE must be in the core of the MPNF game. This proves Part (b). This completes the proof.

16.6 An Example

We now present an example to demonstrate the non-cooperative implementation of the core of the MPNF game. To do this, we will reconsider the example in Fig. 16.4. The characteristic function for this procurement feasibility graph is given in Table 16.2.

The core of this MPNF game is the set of allocations $x = (x_1, x_2, x_3)$ such that the following conditions hold:

1. $x_1 + x_2 + x_3 = 4$
2. $x_i + x_j \geq 0, \quad \forall j \in \{1,2,3\}, \ i \neq j$
3. $x_i \quad \forall j \in \{1,2,3\}$

It is easy to verify that $x = \left(\frac{4}{3}, \frac{4}{3}, \frac{4}{3}\right)$ is one such allocation vector which is in the core of the MPNF game defined by the characteristic function in Table 16.2. We will now see how this allocation corresponds to the sub-game perfect Nash equilibrium of the extensive-form game Γ defined in Sect. 16.5.

In Γ, we have two stages. In Stage 0, every agent i picks an outcome ω_i and a permutation of the agents 1, 2, 3. The outcome ω_i is the status quo outcome that agent i would like to see emerge at the end of Stage 0 and the permutation is the order in which he would like the game to continue in Stage 1.

We let ω_i be the outcome where agent $i(i \in \{1, 2, 3\})$ allows two units of flow in the edges (s, b), (b, c), and (c, t) owned by him. The surplus that is generated in this case is 4. In outcome ω_i, we divide this surplus equally among the agents 1, 2, and 3. That is, we let $x = \left(\frac{4}{3}, \frac{4}{3}, \frac{4}{3}\right)$. Consider now the actions that each of the agents takes in Stage 0 of the game Γ.

i. In Stage 0, every agent i chooses an action $s_i^0 = (\omega, \pi^0)$ where the outcome ω is as defined above and π^0 is the identity permutation.
ii. In Stage 1, every agent i chooses, at every node that he has to make an offer, an action $(S, \omega_S) = (N, \omega)$ with the outcome ω being as defined above.
iii. At every node of Stage 1 where an agent has to respond to a status quo outcome $\tilde{\omega}$ and an offer $(S, \hat{\omega})$, he accepts the offer only if $u_i(\hat{\omega}) > u_i(\tilde{\omega})$.

These actions constitute one strategy profile for the agents 1, 2, and 3. We now need to verify if this strategy profile is an equilibrium strategy profile in which case we can say that the core outcome ω which gives each of the agents a payoff of $1\frac{1}{3}$ is achieved non-cooperatively. To do this, we will check if the actions specified for the agents in each of the stages are a best response.

Since the permutation chosen at the end of Stage 0 is the identity permutation, the sequence of play in

Stage 1 of the game is 1, 2, and 3. Since agent 3 is the last in the sequence, the action that he takes is clearly a sub-game perfect Nash equilibrium starting at the node where he has to either accept or reject the proposal vis-a-vis the status quo. Similarly for agent 2 it is a sub-game perfect Nash equilibrium strategy to reject the proposal given that he knows that it is sub-game perfect for agent 3 to also reject at his decision node. Now, for agent 1 at his decision node in Stage 1, it is clear that if he proposes any outcome which gives any of the agents 2 and 3 less utility than what they would get from the status quo outcome, then it would be rejected. And, if he proposes an outcome which gives agents 2 and 3 a utility payoff that is greater than what they receive from the status quo outcome, then it can come only at the cost of less utility for agent 1. So, the action for agent 1 at his decision node in Stage 1 is sub-game perfect.

Now, in Stage 0, if the agents do not agree upon a status quo outcome then the game ends and each of them gets zero payoff. Compared to this, if they each specify the same status quo outcome ω then they go to Stage 1 of the game and from the discussion above, we know that the specified actions at the decision nodes in this stage of the game constitute a sub-game perfect Nash equilibrium. So, the agents do not have an incentive to deviate from the specified actions in Stage 0 of the game. This means that the strategy profile outlined above is a Nash equilibrium strategy profile. So, we can conclude that the outcome ω which gives each of the agents a payoff $\frac{4}{3}$ can be implemented non-cooperatively through the extensive-form game Γ.

16.7 Conclusion

We conclude by first reiterating the contributions that we have made in this chapter and then pointing out the further course of investigation.

1. We have formulated the multiple unit single item procurement network formation (MPNF) problem as a cooperative game.
2. We have investigated the conditions for the non-emptiness of the core of the MPNF game.
3. We then provided an extensive-form game to implement the core of the MPNF game and showed that the sub-game perfect Nash equilibria of the extensive-form game coincide with the allocations in the core of the MPNF game whenever it is non-empty.

The investigation of the procurement network formation problem for the single item, multiple unit case in this chapter lays the foundation for extending the problem and the analysis in several directions.

First, in this chapter, we have assumed that the buyer just reveals his valuation to the agents in the network and thereafter the agents need to coordinate their actions to generate and share the surplus. Often, the buyer too is interested in bargaining with the suppliers so that he may also get a share of the surplus. In this case, the problem turns out to be a market game whose core is always non-empty and could have a large number of elements (Myerson 1991). In such a case, it is more useful to seek a single valued solution concept since it is not clear which one of the many possible solutions agents will settle upon. The Shapley value is one such solution concept that is widely accepted.

Second, in this chapter, we have assumed that agents have complete information about the cost structure of each of the agents who own edges in the network. It is obvious that there are several situations when this assumption may hold in only a limited way or not at all. For instance, agents are likely to have complete information about adjacent edges in the network but may have no information about other edges. The question now is whether we can devise a strategy-proof scheme where we do not have to make high incentive payments in order to get a complete picture of the state of the network before we select a flow and make allocations of surplus. In other instances, edges may be completely uninformed about the cost structures of other edges. The question then is to develop and analyze an appropriate cooperative model which describes this situation.

References

Aumann RJ (1987) Game theory. In: Eatwell J, Milgate M, Newman P (eds) The new palgrave, a dictionary of economics. Macmillan, London, pp 460–482

Babaioff M, Nisan N (2004) Concurrent auctions across the supply chain. J Artif Intell Res 21:595–629

Babaioff M, Walsh WE (2005) Incentive-compatible, budget-balanced, yet highly efficient auctions for supply chain formation. Decis Support Syst 39:123–149

Bajari PL, McMillan RS, Tadelis S (2003) Auctions versus negotiations in procurement: an empirical analysis. Technical report, NBER Working paper series, Department of Economics, Stanford University

Biswas S, Narahari Y (2004) Object oriented modeling for decision support in supply chain networks. Eur J Oper Res 153:704–726

Bondereva ON (1963) Some applications of linear programming methods to the theory of cooperative games. Problemy Kibernetiki 10:119–139 (in Russian)

Cachon P, Netessine S (2004) Game theory in supply chain analysis. In: Simchi Levi D, Wu DS, Shen ZJ (eds) Handbook of quantitative supply chain analysis, modelling in the e-business era. Kluwer Academic Publishers, Boston

Collins J, Gini M (2001) A testbed for multi-agent contracting for supply chain formation. In: IJCAI-2001 workshop on artificial intelligence and manufacturing

Collins J (2002) Solving combinatorial auctions with temporal constraints in economic agents. Ph.D. thesis, Department of Computer Science and Engineering, University of Minnesota

Erenguc SS, Simpson NC, Vakharia AJ (1999) Integrated production/distribution planning in supply chains: an invited review. Eur J Oper Res 115:219–236

Fan M, Stallert J, Whinston AB (2003) Decentralized mechanism design for supply chain organizations using an auction market. Inf Syst Res 14(1):1–22

Garg D, Narahari Y, Viswanadham N (2004) Design of six sigma supply chains. IEEE Trans Autom Sci Eng 1(1):38–57

Garg D (2006) Design of innovative mechanisms for contemporary game theoretic problems in eletronic commerce. Ph.D. thesis, Department of Computer Science and Automation, Indian Institute of Science

Jiong S, Sadeh N (2004) Coordinating multi-attribute procurement auctions subject to finite capacity considerations. Technical report, School of Computer Science, Carnegie Mellon Unviversity, CMU-ISRI-03-105 Tech. Report

Leng M, Parlar M (2005) Game theoretic applications in supply chain management: a review. Inf Syst Oper Res 43(3):187–220

Mas-Collel A, Whinston MD, Green JR (1995) Microeconomic theory. Oxford University Press, New York

Maskin E (1985) The theory of implementation in Nash equilibrium: a survey. In: Hurwicz L, Schmiedler D, Sonnenschien H (eds) Social goods and social organization. Cambridge University Press, Cambridge

McAfee R (1992) A dominant strategy double auction. J Econ Theor 56:434–450

Moore J (1992) Implementation, contracts and renegotiation in environments with complete information. In: Laffont JJ (ed) Advances in economic theory. VI world congress of the econometric society. Cambridge University Press, Cambridge

Myerson RB (1991) Game theory: analysis of conflict. Harvard University Press, Cambridge

Nagarajan M, Sosic G (2008) Game-theoretic analysis of cooperation among supply chain agents: review and extensions. Eur J Oper Res 187(3):719–745

Serrano R, Vohra R (1997) Non-cooperative implementation of the core. Soc Choice Welf 14:513–525

Shapley LS (1967) On balanced sets and cores. Nav Res Logist Q 14:453–460

Thomson W (2005) Divide-and-permute. Games Econ behav 52:186–200

Viswanadham N, Goankar RS (2003) Partner selection and synchronized planning in dynamic manufacturing networks. IEEE Trans Robot Autom 19(1):117–130

Walsh WE, Wellman MP (1999) Efficiency and equilibrium in task allocation economies with hierarchical dependencies. In: Sixteenth international joint conference on artificial intelligence, pp 520–526

Walsh WE, Wellman MP, Ygge F (2000) Combinatorial auctions for supply chain formation. In: Proceedings of ACM conference on electronic commerce (EC-00), pp 260–269

Walsh WE, Wellman MP (2003) Decentralized supply chain formation: a market protocol and competitive equilibrium analysis. J Artif Intell 19:513–567

Wu D (2004) Supply chain intermediation: A bargaining theoretic framework. In: Simchi Levi D, Wu DS, Shen ZJ (eds) Handbook of quantitative supply chain analysis, modelling in the e-business era. Kluwer Academic Publishers, Amsterdam

Index

L. Benyoucef et al. (eds.), *Applications of Multi-Criteria and Game Theory Approaches*, Springer Series in Advanced Manufacturing, DOI: 10.1007/978-1-4471-5295-8,